
Desarrollo de un entorno de simulación en python-Qt, para USVs y barreras de contención de vertidos marinos.

Author:

Luis Antonio SAAVEDRA
PALACIOS

Supervisor:

Dr. Juan JIMENEZ
CASTELLANOS



Abstract and keywords

The problem of marine dumping is becoming more and more worrisome. There are many techniques to collect marine spills, in most of these are used marine spill containment barriers, to be able to tow and take them to a place to pick it up. The deployment of the barriers depends on climatic factors, marine currents and the direction of the wind, as well as the maneuver of the boat that will drag them. On the other hand, the use of unmanned vehicles or navigation robots, coupled with these barriers for the towing of marine spills, is an interesting alternative. A computer tool was created to simulate the action of drag barriers attached to unmanned surface vehicles and thus have previous simulations before doing real tests, this project is the realization of a graphical interface of this tool to facilitate their use.

- Keywords

Booms, USVs, towing of barriers, deployment, maneuver, interface of ship and chain.

Capítulo 1

Resumen y palabras clave

El problema de los vertidos marinos está siendo cada vez más preocupante. Existen muchas técnicas para recoger los vertidos marinos, en la mayoría de éstas se utilizan barreras de contención de vertidos marinos, para poder remolcarlos y llevarlos a un sitio en el que recogerlo. El despliegue de las barreras, depende de factores climáticos, corrientes marinas y la dirección del viento, así como también de la maniobra del barco que los arrastrará. Por otro lado el uso de vehículos no tripulados o robots de navegación, unido a éstas barreras para el remolque de vertidos marinos, es una alternativa interesante.

Una herramienta informática se creó para simular la acción de arrastrar barreras unido a vehículos de superficie no tripulados y así tener simulaciones previas antes de hacer pruebas reales, este proyecto es la realización de una interfaz gráfica de ésta herramienta para facilitar su uso.

- Palabras clave

Barreras de contención de vertidos marinos, vehículos de superficie no tripulados, remolque de barreras, despliegue, maniobra, interfaz de barco y cadena.

Índice general

1. Resumen y palabras clave	5
Resumen	5
Lista de figuras	9
2. Introducción	11
Introducción	11
3. Ecuaciones del Modelo	17
Ecuaciones del Modelo	17
3.1. Ecuaciones del USV	17
3.2. Ecuaciones de la Barrera	18
3.3. Cables de remolque	22
3.4. Control Cooperativo	24
4. Descripción de la herramienta de simulación	25
Descripción de la herramienta de simulación	25
5. Descripción de la interfaz	27
Descripción de la interfaz	27
5.1. Interfaz de la clase barco y cadena	28
5.2. Interfaz Principal	30
5.3. Interfaz del botón Editar Barcos y Cadenas	32
5.4. Interfaz de Combinar	35
5.5. Interfaz del botón Simular Barco y Cadena	37
6. Simulación Completa	49
Simulación Completas	49
7. Implementación de la Interfaz.	57
Implementación de la Interfaz	57

8. Conclusión y futuros trabajos	61
Conclusión y futuros trabajos	61

Índice de figuras

2.1. vertido de petroleo	11
2.2. Dos barcos remolcando barreras de vertidos marinos	12
3.1. Diseño del USV mostrando los sistemas de referencia utilizados en el modelo	19
3.2. Elementos principales del diseño de la Barrera	21
3.3. Control cooperativo del rumbo	23
3.4. Control cooperativo de la velocidad	23
4.1. Listado de atributos de la clase barco y cadena	26
5.1. Interfaz de la clase barco	28
5.2. Interfaz de la clase cadena	29
5.3. Ampliando la explicación del atributo kwpid[1]	29
5.4. Error al querer crear dos barcos con el mismo identificador . .	30
5.5. interfaz principal	31
5.6. interfaz de Edición de Barco o Cadena	32
5.7. Dos pantallas de la interfaz Editar Barcos y Cadenas	33
5.8. Muestra el botón añadido : Mostrar Dibujo Barco	33
5.9. El dibujo del barco(USV)	34
5.10. Interfaz del botón Combinar	36
5.11. Interfaz del botón Combinar cuando se escoge el barco y la cadena	36
5.12. interfaz del botón Elongación	39
5.13. dos ventanas del botón Radio Link, en donde se puede observar que hemos escogido los atributos ide y theta	40
5.14. Interfaz del botón Simular Barco y Cadena	41
5.15. Simulación con 0.001 de paso de integración y una consigna rumbo de 45 grados	42
5.16. Parado el proceso de carga de datos	43
5.17. Cargando datos para la simulación	44

5.18. Eligiendo un grupo de iteraciones	45
5.19. Elegimos la iteración 60 y luego de hacer <i>clíc</i> se muestra el esquema de ésta iteración	46
5.20. Se muestra la elección de la gráfica Rumbo y la gráfica Rumbo	47
6.1. Dos USVs arrastrando una barrera de 200 metros unidos me- diante un cable de 5 metros, una consigna de rumbo de 45 grados y una velocidad final de $1m/s$ y un tiempo de 10 se- gundos de simulación.	52
6.2. Gráficas de la simulación completa	53
6.3. Dos USVs arrastrando una barrera de 200 metros unidos me- diante un cable de 5 metros, a partir de una consigna de rumbo de 45 grados hasta llegar a una consigna de rumbo de 90 grados y una velocidad final de $1m/s$, en un tiempo de 10 segundos de simulación.	55
6.4. Gráficas de la siguiente simulación completa siguiendo la si- mulación completa anterior	56

Capítulo 2

Introducción

En el año 2018 se lanzó la primera operación mundial contra los vertidos marinos llamada '30 días en el mar' que fue coordinada por la Europol y dirigida por la Interpol. Los distintos vertidos marinos son una preocupación cada vez mayor, por parte de los Gobiernos del mundo (información sacada de [4]).

Un derrame de petróleo o marea negra es un vertido que se produce por accidente o práctica inadecuada que contamina el medio ambiente y perjudica gravemente a la fauna y a la pesca. En la Figura 2.1 se muestra una imagen de un derrame de petróleo.

Cuando ocurre un derrame de petróleo se toman varias medidas, por ejemplo



Figura 2.1: vertido de petróleo

quemar el vertido, si se está en mar abierto, o usar productos químicos para descomponer el vertido y así hacer que desaparezca. Pero éstas medidas tras-

ladan el problema del vertido a otro entorno (del agua al aire por ejemplo al quemar el petróleo). Por ello los gobiernos utilizan medidas mecánicas, dentro de éstas medidas se destaca el uso de barreras y skimmers (información sacada de [5]). En este sentido cuando ocurre un derrame de petróleo en mar abierto, desplegar las barreras para rodear todo el vertido conectando éstas a dos barcos, de manera que se encapsule el vertido y se pueda arrastrar conjuntamente y llevarlo a un punto para que se pueda usar un skimmer y separar el vertido del mar. Es una tarea dura y pesada que requiere de una tripulación bien entrenada. Debido a esto ante emergencias de derrame, el problema es que no siempre la tripulación está lista para este tipo de acción, ya que la tripulación no puede trabajar indefinidamente y sin descanso. En la Figura 2.2 se muestra a dos barcos arrastrando una barrera de contención de vertidos marinos.



Figura 2.2: Dos barcos remolcando barreras de vertidos marinos

El creciente desarrollo de vehículos autónomos en la última década, nos hace imaginar el despliegue y arrastre de barreras de manera automática, utilizando vehículos de superficie autónomos (USVs) de ésta forma se solucionaría el problema mencionado anteriormente. Replicando dichos barcos y teniendo un sistema de repostado se podría tener a estos trabajando de manera continua. Algunos experimentos reales de vehículos de superficie autónomos unido a barreras para el remolque de vertidos marinos se encuentran en el trabajo de Jose M. Giron-Sierra [2].

Desde el punto de vista de sistemas de control, los escenarios de despliegue y remolque automático de las barreras (Booms) mediante vehículos de super-

ficie autónomos(USVs) resultan interesantes y desafiantes. Podemos considerar sucesivas capas para describir estos escenarios de manera más sencilla. Una primera capa se encargaría de diseñar vehículos de superficie autónomos(USV's) que sean capaces de remolcar barreras. Una capa de control básica se encargaría del control de la velocidad, rumbo y el auto-guiado. El motor y la dirección del USV deberían tener suficiente potencia para moverse y maniobrar bajo la tensión que ejercen las barreras que está arrastrando. En el caso de más de un USV involucrado en la operación de remolque de las barreras, tendríamos que tener un sistema para coordinar sus maniobras de lo contrario se podrían quedar fuera de control y se romperían las barreras(información sacada del trabajo de Juan Jimenez [6]). Un enfoque sugiere aplicar 'La Estrategia de Control Cooperativo' descrito en el trabajo de Filippo Arrichiello [1]. Sería muy conveniente tener una capa de planificación para poder definir la mejor o la peor trayectoria que va a recorrer la barrera desplegada, cuando los USVs remolcan algún vertido. Ésta capa podría estar implementada en todos los USVs o en una base central la cual daría órdenes a los USVs involucrados.

Ahora bien ¿Como transferimos la experiencia y el buen entrenamiento de una tripulación a un vehículo de superficie autónomo(USV)?. Podríamos considerar el uso de simulaciones para reproducir el despliegue de barreras de vertidos marinos unido a USVs. Éstas simulaciones nos harían conocer qué capas están presentes en el movimiento del USV junto con las barreras y cuantificar la tensión presente al remolcar dichas barreras. También éstas simulaciones nos ayudarían a conocer las estrategias de control necesarias así como también a mejorar el diseño del USV y a escoger la barrera adecuada(información sacada del trabajo de Juan Jimenez [6]).

Introduction

In the year 2018 launched the first global operation against Marine dumping called '30 days at sea' which was coordinated by Europol and directed by Interpol. The various marine discharges are an increasing concern, by the Governments of the world(see information at [4]). An oil spill or black tide is a spill that occurs because of an accident or inadequate practice that pollutes the environment and seriously harms wildlife and fishing. Figure 2.1 shows an image of an oil spill. When an oil spill occurs, several measures are taken, such as burning the spill, if it is in the open sea, or using chemicals to break down the spill and make it disappear. But these measures move the problem of dumping to another environment (from water to air for example when burning oil). That is why governments use mechanical measures, within these measures the use of barriers and skimmers is highlighted(see information at [5]). In this sense when an oil spill occurs in the open sea, deploy the barriers to surround all the spill connecting these to two boats, so that the spill is encapsulated and can be dragged together and taken to a point so that you can use a Skimmer and separate the spill from the sea. It is a hard and heavy task that requires a well-trained crew. Because of this in the face of spill emergencies, the problem is that the crew is not always ready for this type of action, as the crew cannot work indefinitely and relentlessly. Figure 2.2 shows two boats dragging a barrier containing marine spills. The growing development of autonomous vehicles in the last decade, makes us imagine the deployment and drag of barriers automatically, using autonomous surface vehicles (USVs) in this way would solve the problem mentioned above. By replicating these boats and having a refueling system you could have these working on a continuous basis. Some real experiments of autonomous surface vehicles coupled with barriers for the trailer of marine spills are in the work of Jose M. Giron-Sierra [2].

From the point of view of control systems, the scenarios of deployment and automatic trailer of the barriers (Booms) by means of self-contained surface vehicles (USVs) are interesting and challenging. We can consider successive layers to describe these scenarios in a simpler way. A first layer

would be responsible for designing self-contained surface vehicles(USVs) that are capable of towing barriers. A basic control layer would be responsible for speed, direction and self-guiding control. The motor and the direction of the USV should have enough power to move and maneuver under the tension exerted by the barriers it is dragging. In the case of more than one USV involved in the towing operation of the barriers, we would have to have a system to coordinate their maneuvers otherwise they could be left out of control and the barriers would break(see information at the paper of Juan Jimenez [6]). One approach suggests applying 'the cooperative Control strategy' described in the work of Filippo Arrichiello [1]. It would be very convenient to have a planning layer to be able to define the best or worst trajectory that will traverse the deployed barrier, when the USVs tow some spill. This Layer could be implemented in all USVs or in a central base which would give orders to the USVs involved. But how do we transfer the experience and good training of a crew to a self- contained surface vehicle (USV)? We could consider the use of simulations to replicate the deployment of marine dumping barriers linked to USVs. These simulations would make us know what layers They are present in the movement of the USV along with the barriers and quantify the tension present when towing these barriers. Also these simulations would help us to know the necessary control strategies as well as to improve the design of the USV and to choose the appropriate barrier(see information at the paper of Juan Jimenez [6]).

Capítulo 3

Ecuaciones del Modelo

En ésta parte describiremos brevemente las ecuaciones necesarias para modelar el movimiento de los vehículos autónomos de superficie(USV) y las barreras de vertidos marinos(Booms), al momento de realizar la acción de remolque de éstas barreras por dichos vehículos para luego describir la interfaz y su implementación. Se puede ver de manera más detallada la descripción de las ecuaciones en el trabajo de Juan Jimenez [3].

3.1. Ecuaciones del USV

Usaremos para el USV un modelo simplificado de dos dimensiones, la longitud(o la eslora) del USV será de l_s , de masa m_b y momento de inercia I_b . Estará dotado de un modelo dinámico que considera una fuerza de propulsión genérica F_m aplicada a su centro de masa y se le aplicará un momento genérico M_b . Consideramos la resistencia opuesta ejercida por el agua al movimiento del USV. La cual descomponemos en dos terminos, uno es proporcional a la velocidad del USV y el otro es proporcional al cuadrado de ésta velocidad. Consideramos los coeficientes μ_l y μ_{l2} . Los cuales tienen en cuenta, la contribución del barco a la resistencia en dirección de la longitud del USV. Consideramos la resistencia la cual modelamos de manera proporcional a la longitud del USV y a un par de coeficientes llamadas μ_t y μ_{t2} . Las masas añadidas m_A las modelamos de manera simplificada siguiendo un enfoque similar al que empleamos para la resistencia. Usamos un coeficiente constante para representar la masa añadida en la dirección de avance del USV y otro para representar la masa añadida en la dirección transversal. Los efectos de las masas añadidas en el giro del barco se desprecian. Consideramos las Fuerzas externas aplicadas al USV por las corrientes marinas, al restar la velocidad del agua de la velocidad del USV, y usamos ésta velocidad

relativa en términos de la resistencia descrita en el párrafo anterior. Además consideramos la existencia de una única fuerza externa F_e , aplicada al casco del barco y su correspondiente momento M_e . Ésta fuerza se puede utilizar para modelar la existencia de una tensión externa, debida a la presencia de dispositivos remolcados por el USV. En la Figura 3.1 mostramos los sistemas de referencia de coordenadas (x,y), y las fuerzas empleadas en la dinámica del USV. A continuación mostramos las ecuaciones para obtener los componentes de la aceleración en plano (x,y) y la aceleración angular, utilizando las fuerzas y el momento aplicado al USV.

$$\begin{aligned}
& (m_b + m_{A\parallel} \cos(\theta_b) - m_{A\perp} \sin(\theta_b)) \cdot a_{bx} = \\
& F_{mx} - \mu_l [v_{bx} \cos(\theta_b) + v_{by} \sin(\theta_b)] \cos(\theta_b) \\
& \quad + \mu_t l_s [v_{by} \cos(\theta_b) - v_{bx} \sin(\theta_b)] \sin(\theta_b) \\
& \quad - \mu_{l2} l_s [v_{bx} \cos(\theta_b) + v_{by} \sin(\theta_b)] |\vec{v}_b| \cos(\theta_b) \\
& + \mu_{t2} l - s [v_{by} \cos(\theta_b) - v_{bx} \sin(\theta_b)] |\vec{v}_b| \sin(\theta_b) + F_{ex}
\end{aligned} \tag{3.1}$$

$$\begin{aligned}
& (m_b + m_{A\parallel} \sin(\theta_b) + m_{A\perp} \cos(\theta_b)) \cdot a_{by} = \\
& F_{my} - \mu_l [v_{bx} \cos(\theta_b) + v_{by} \sin(\theta_b)] \sin(\theta_b) \\
& \quad - \mu_t l_s [v_{by} \cos(\theta_b) - v_{bx} \sin(\theta_b)] \cos(\theta_b) \\
& \quad - \mu_{l2} l_s [v_{bx} \cos(\theta_b) + v_{by} \sin(\theta_b)] |\vec{v}_b| \sin(\theta_b) \\
& - \mu_{t2} l - s [v_{by} \cos(\theta_b) - v_{bx} \sin(\theta_b)] |\vec{v}_b| \cos(\theta_b) + F_{ey}
\end{aligned} \tag{3.2}$$

$$I_b \alpha_b = M - \mu_a l_s \omega_b - \mu_{a2} l_s \omega_b |\omega_b| + M_e \tag{3.3}$$

3.2. Ecuaciones de la Barrera

Las ecuaciones de las barrera de vertidos marinos las derivamos de un modelo simplificado de 'eslabón barrera'. El modelo considera que la barrera está formada por eslabones rígidos unidos entre sí. Los eslabones se unen para formar la barrera al completo. Éstas pueden girarse y moverse siempre que sigan unidas entre sí. En la Figura 3.2 mostramos la Barrera donde cada eslabón se representa como segmentos concatenados, y cada punto representa el centro de masa de cada eslabón. También mostramos en la Figura 3.2 los vectores unitarios normales \vec{n}_i y los vectores unitarios paralelos \vec{p}_i empleados en las ecuaciones de la barrera y por último los vectores \vec{r}_i que representan el

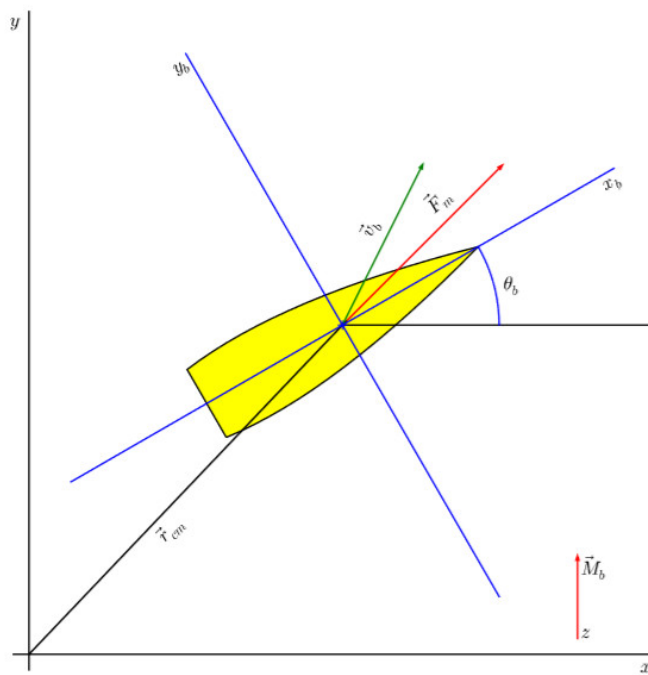


Figura 3.1: Diseño del USV mostrando los sistemas de referencia utilizados en el modelo

vector posición del centro de masa de los eslabones. consideramos que cada eslabón está bajo la acción de las tensiones aplicadas a sus extremos. Consideramos las fuerzas resistivas como en el caso del USV descrito anteriormente. Las masas añadidas son consideradas en dirección normal al eslabón. En el plano (x,y) la masa total del eslabón está representado por la diagonal de la matriz:

$$m_{Ai} = \begin{pmatrix} m_l + m_{la} \cos \theta_i & 0 \\ 0 & m_l + m_{la} \sin \theta_i \end{pmatrix} \quad (3.4)$$

Las siguientes ecuaciones describen la dinámica de cada eslabón, describiendo así todos los eslabones la dinámica de la barrera al completo.

$$T_{i,i+1}^{\vec{}} - T_{i-1,i}^{\vec{}} - F_{ri} = m_{Ai} \vec{a}_i \quad (3.5)$$

$$(T_{i,i+1}^{\vec{}} \vec{n}_i - T_{i-1,i}^{\vec{}} \vec{n}_i)l - M_{ri} = I\alpha_i \quad (3.6)$$

F_{ri} representa las Fuerzas de resistencia lineal y cuadrática

$$\vec{F}_{ri} = (|\vec{v}_i \cdot \vec{n}_i|s + \vec{v}_i \cdot \vec{p}_i|q) \frac{\vec{v}_i}{|\vec{v}_i|} + (|\vec{v}_i \cdot \vec{n}_i|s2 + \vec{v}_i \cdot \vec{p}_i|q2)\vec{v}_i \quad (3.7)$$

M_{ri} representa el momento cuadrático y lineal de la fuerza de resistencia

$$M_{ri} = A_1\omega_i + A_2\omega_i|\omega_i| \quad (3.8)$$

En el eslabón final izquierdo

$$T_{1,2}^{\vec{}} - F_{left}^{\vec{}} - F_{r1}^{\vec{}} = m_{A1} \cdot \vec{a}_1 \quad (3.9)$$

$$(T_{1,2}^{\vec{}} \cdot \vec{n}_1 + F_{left}^{\vec{}} \cdot \vec{n}_1)l - A_{\omega_1} = I\alpha_1 \quad (3.10)$$

En el eslabón final derecho

$$F_{right}^{\vec{}} - T_{m-1,m}^{\vec{}} - F_{rm}^{\vec{}} = m_{Am} \cdot \vec{a}_m \quad (3.11)$$

$$(F_{right}^{\vec{}} \cdot \vec{n}_m + T_{m-1,m}^{\vec{}} \cdot \vec{n}_m)l - M_{ri} = I\alpha_m \quad (3.12)$$

Añadimos la ecuación que establece la condición de que cada eslabón debe permanecer conectado con los eslabones adyacentes.

$$\vec{r}_i - l\vec{p}_i - lp_{i+1}^{\vec{}} - r_{i+1}^{\vec{}} = 0 \quad (3.13)$$

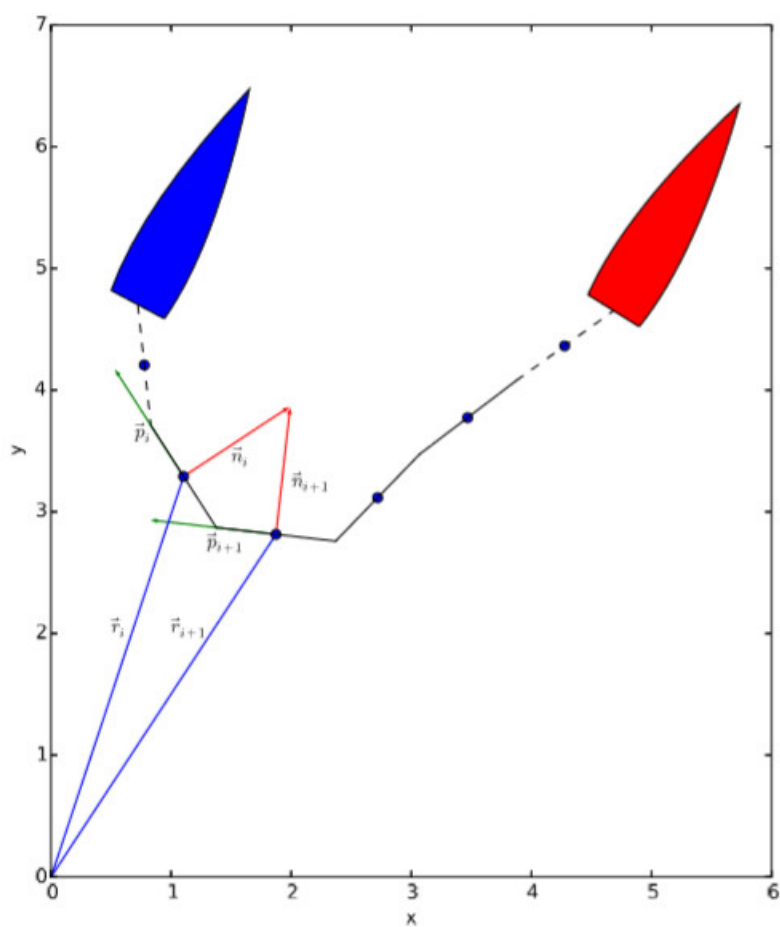


Figura 3.2: Elementos principales del diseño de la Barrera

$$\vec{a}_i + l\vec{p}_i\omega_i^2 + l\vec{n}_i\alpha_i + lp_{i+1}\vec{\omega}_{i+1}^2 + ln_{i+1}\alpha_{i+1} - a_{i+1} = 0 \quad (3.14)$$

Las ecuaciones (5),(6),(9),(10),(11),(12) y (14) forman un conjunto completo de ecuaciones y es posible resolverla numericamente para obtener el movimiento de la barrera.

3.3. Cables de remolque

Usamos cables de remolque para realizar el remolque de las barreras usando USVs. Consideramos la tensión en el cable como función de su estiramiento $\Delta d = d - d_0$ siendo d_0 su distancia inicial.

$$T_{cbl} = \begin{cases} 0 & \text{si } \Delta d < 0 \\ \frac{E}{d_0}\Delta d & \text{si } \Delta d \geq 0 \end{cases} \quad (3.15)$$

En donde E es el módulo de Young o módulo de elasticidad longitudinal. Las siguientes ecuaciones describen la tensión del cable cuando está unido al USV y a la barrera, durante la acción del remolque de las barreras.

$$\begin{aligned} d_l &= |\vec{r}_{cm} - \frac{\vec{l}_s}{2} - \vec{r}_1 - \vec{p}_1|^2, \text{ SoL (USV a la izquierda)} \\ d_r &= |\vec{r}_m - \vec{p}_m - \vec{r}_{cm} + \frac{\vec{l}_s}{2}|^2, \text{ SoR (USV a la derecha)} \end{aligned} \quad (3.16)$$

$$\begin{aligned} u_{cblr} &= \frac{\vec{r}_{cm} - \frac{\vec{l}_s}{2} - \vec{r}_1 - \vec{p}_1}{d_l}, \text{ SoL} \\ u_{cbl} &= \frac{\vec{r}_m - \vec{p}_m - \vec{r}_{cm} + \frac{\vec{l}_s}{2}}{d_r}, \text{ SoR} \end{aligned} \quad (3.17)$$

$$\begin{aligned} \vec{F}_e &= T_{cbl} \cdot u_{cbl} \text{ SoL,} \\ F_{left} &= T_{cbl} \cdot u_{cbl} \\ \vec{F}_e &= T_{cblr} \cdot u_{cblr} \text{ SoR,} \\ F_{right} &= T_{cblr} \cdot u_{cblr} \end{aligned} \quad (3.18)$$

Heading control

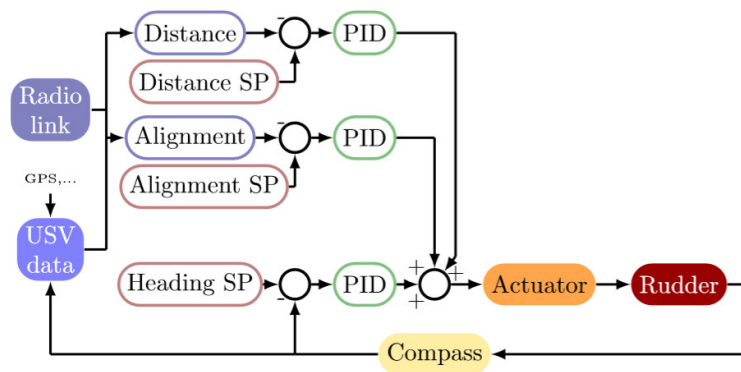


Figura 3.3: Control cooperativo del rumbo

Speed control

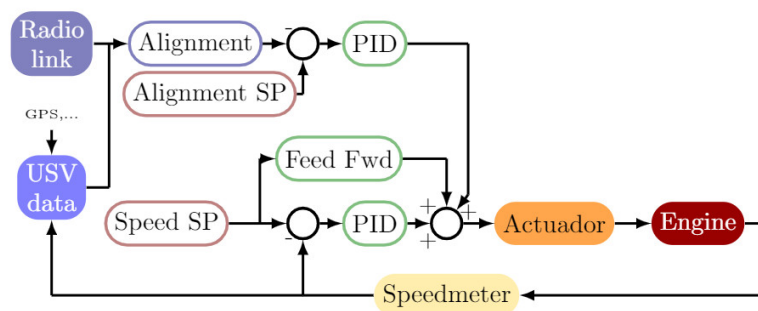


Figura 3.4: Control cooperativo de la velocidad

3.4. Control Cooperativo

Cuando dos barcos remolcan una barrera es necesario que estén coordinados. En el caso de un remolque automático llevado a cabo por USVs la coordinación se puede lograr aplicando estrategias de control cooperativo. Las Figura 3.3 y Figura 3.4 representan los sistemas de control de rumbo y velocidad a bordo de un modelo USV. Ambos sistemas de control actúan usando información suministrada por la pareja en cuestión. Los datos se recogen desde un enlace de radio. Este enfoque resulta útil para realizar maniobras básicas.

Capítulo 4

Descripción de la herramienta de simulación

Las ecuaciones del (1) al (18) describen la evolución temporal de los estados de los USVs y las barreras siempre que se conozcan las condiciones iniciales. La integración numérica de éstas ecuaciones es la parte más importante del sistema de simulación. Partiendo de alguna configuración en un tiempo inicial t_0 y seleccionando un paso de integración Δt las ecuaciones del modelo se integran paso a paso hasta un tiempo de integración final t_f . Tomando en cuenta lo anterior se desarrolló una herramienta en Python. La simulación se basa en dos clases básicas: La clase barco y la clase cadena. La clase barco tiene tres procedimientos principales:

- movimiento(.)
- controlador(.)
- propulsión(.)

La clase cadena tiene dos procedimientos principales:

- movimiento(.)
- mtrs (.)

Las simulaciones son ejecutadas siguiendo una filosofía de scripting. El código está escrito en scripts en donde se definen las clases barco, cadena y sus respectivas funciones. También se usan plantillas de script que se adaptan en cada caso a las especificaciones de algún escenario en concreto (información sacada del trabajo de Juan Jimenez [6]). El código está disponible en Github

26CAPÍTULO 4. DESCRIPCIÓN DE LA HERRAMIENTA DE SIMULACIÓN

de Juan Jimenez (<https://github.com/juanjimenez/pyships>). Como vemos en la Figura 4.1, para crear la clase barco tenemos cierta cantidad importante de atributos que tendríamos que editar cada vez que queramos hacer algún cambio en nuestra posible simulación y también tendríamos que conocer como está estructurado el script. Por ello para ayudar al usuario al uso de ésta herramienta se creó una interfaz que mejora y simplifica el uso de la misma. La descripción de la interfaz e implementación de la misma se explican en los próximos capítulos.

USV parameters:			boom parameters		
ide	:identification number	1	esl	:number of elements	10
thewjmax	:Maximum water-jet/rudder turn angle	0.52	s	:element surface	2.0
Ac	:Distance from the rudder axis to CoM	10.0	s2	:element thick	0
mb	:USV mass	1000.0	q	:damping coefficient (transverse)	0.1
mA	:added mass	[0. 0.]	q2	:damping quadratic coefficient (transverse)	0
mul	:damping linear coefficient (surge)	1000.0	A	:damping coefficient (turning)	10.0
mul2	:damping quadratic coefficient (surge)	0.0	Anch	:Tip(s) anchored to the dock	[0. 0.]
Ib	:Inertia moment	1000.0	L	:half length of an element	0.5
mua	:damping coefficient (yaw)	1000.0	m	:element mass	1.0
mua2	:damping quadratic coefficient (yaw)	0.0	mA	:element added mass coef. (transverse)	0
ls	:length	2.0	Fd	:force applied to the boom right end	[0. 0.]
mut	:damping linear coefficient (sway)	10000.0	Fi	:force applied to the boom left end	[0. 0.]
mut2	:damping quadratic coefficient (sway)	0.0	I	:moment of inertia	0.08333
pmax	:maximum engine power	36750.0	mtr_s	:method to calculate the strains matrix	<bound method cadena.mtrb_s of <boom.cadena instance at 0x0B884710>>
Tsf	: time constant of the USV engine	0.1	Variables and their meaning (values are omitted)		
Tsw	: time constant of USV guiding system	0.1	normal	:unit vector normal to the elements	
tipo	: type of USV propulsion system	waterjet	para	:unit vector parallel to the element	
cfr	: form factor for rudder guided USVs	100	cms	:elements center of mass	
variables and their meaning (values are omitted)			v	:elements speed	
pb	:position		vr	:element speed ref. water	
vb	:speed		vmod	:element speed norm	
vr	:speed relative to water		a	:element acceleration	
ab	:acceleration		per	:element external perturbation	
alfab	:angular acceleration		alfa	:element angular acceleration	
wb	:angular speed		w	:element angular speed	
theta	:heading		T	:strains	
Fm	:exerted force		M	:strain matrix	
setl	:engine setpoint		B	:Independent terms to calculate T, from $M \cdot T = B$	
setw	:heading setpoint				
thewj	:waterjet/rudder orientation				
M	:moment applied to the USV				
tiempo	:USV simulation time				

Figura 4.1: Listado de atributos de la clase barco y cadena

Capítulo 5

Descripción de la interfaz

Qt es un entorno de trabajo orientado a objetos que se utiliza para desarrollar programas que utilicen interfaz gráfica de usuario (información sacada de [7]). PyQt es una referencia a la biblioteca gráfica Qt para el lenguaje de programación Python (información sacada de [8]). En nuestro caso utilizaremos PyQt4 por:

1. El uso de las librerías importantes de Python:
 - SciPy
Es una biblioteca libre y de código abierto para Python, que se compone de módulos para la integración numérica, además de módulos para el procesamiento de señales, interpolación y funciones especiales (información sacada de [9]).
 - numPy
Es una extensión de Python, que constituye una biblioteca de funciones matemáticas, para realizar operaciones algebraicas sobre vectores y matrices (información sacada de [10]).
 - Matplotlib
Es una biblioteca de Python, para la generación de gráficos a partir de datos estructurados (array, listas) (información sacada de [11]).
2. Unido al lenguaje multiplataforma Qt, que es ampliamente usado, algunas plataformas que la utilizan son: Google Earth, Virtualbox, Skype.
3. Y por último se puede utilizar parte del código en Python desarrollado en la implementación de la herramienta.

5.1. Interfaz de la clase barco y cadena

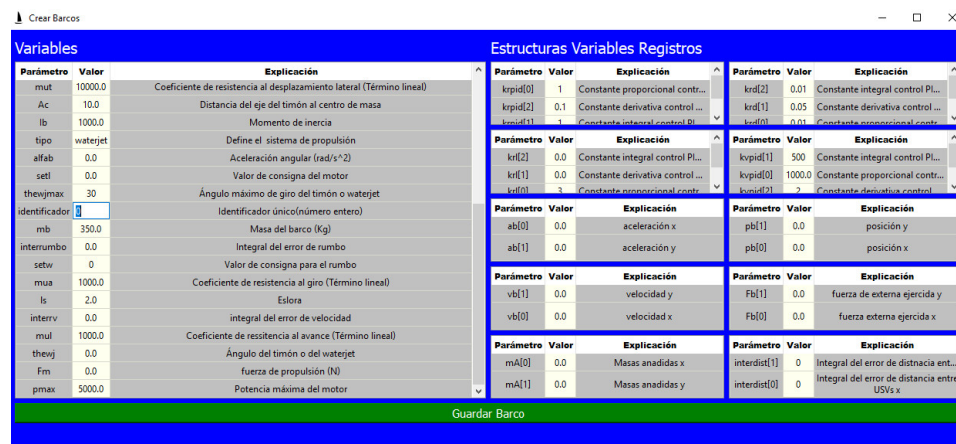
Se empezó creando la interfaz para las clases barco y cadena, ya que como se explicó en el capítulo 4, para hacer simulaciones se deben de usar éstas clases. En la Figura 4.1 vemos que la cantidad de atributos que se tienen que mostrar para cada clase son muchos. Debido a esto se diseñó una interfaz que muestra estos atributos ordenados de tal manera que el usuario puedan verlos al completo.

Como se ve en la Figura 5.1 y 5.2, las interfaces de la clase barco y cadena muestran todos los atributos en un total de 11 y 5 tablas respectivamente. Como se puede observar en éstas Figuras(5.1 y 5.2) en cada tabla, las filas representan los atributos de la clase.

La primera columna muestra el nombre del atributo, en la segunda columna se muestra el valor inicial de cada atributo, el cual se puede editar haciendo *clic* en éste y escribiendo el nuevo valor que se le quiere dar, en la tercera columna se muestra la descripción del atributo.

Las tablas en las Figuras(5.1 y 5.2) están ordenadas de manera que el usuario tenga la máxima visualización de los atributos de cada clase para su uso.

Como vemos en la Figura 5.3 para la interfaz de la clase barco, si no vemos bien la descripción de un atributo(tercera columna de la tabla en donde está el atributo en cuestión). Ésta descripción se puede ampliar haciendo *clic* izquierdo, después de este *clic* se abre una ventana con la descripción del atributo.



The screenshot shows a software interface titled 'Crear Barcos' with a 'Variables' section. It contains a table with three columns: 'Parámetro', 'Valor', and 'Explicación'. The table lists various parameters for a ship, such as 'mut' (lateral resistance coefficient), 'Ac' (distance from center of mass to rudder), 'lb' (moment of inertia), 'tipo' (propulsion system), 'alfab' (angular acceleration), 'setl' (motor command value), 'thetjmax' (maximum rudder angle), 'identificador' (unique identifier), 'mb' (ship mass), 'interrumbo' (heading error integral), 'setw' (rudder command value), 'msa' (yaw resistance coefficient), 'ls' (slur), 'interv' (velocity error integral), 'mul' (advance resistance coefficient), 'thetj' (rudder angle), 'Fm' (propulsion force), and 'pmax' (maximum motor power). Each parameter has a numerical value and a descriptive text.

Parámetro	Valor	Explicación
mut	10000.0	Coefficiente de resistencia al desplazamiento lateral (Término lineal)
Ac	10.0	Distancia del eje del timón al centro de masa
lb	1000.0	Momento de inercia
tipo	waterjet	Define el sistema de propulsión
alfab	0.0	Aceleración angular (rad/s ²)
setl	0.0	Valor de consigna del motor
thetjmax	30	Ángulo máximo de giro del timón o waterjet
identificador	0	Identificador único(número entero)
mb	350.0	Masa del barco (Kg)
interrumbo	0.0	Integral del error de rumbo
setw	0	Valor de consigna para el rumbo
msa	1000.0	Coefficiente de resistencia al giro (Término lineal)
ls	2.0	Eslera
interv	0.0	Integral del error de velocidad
mul	1000.0	Coefficiente de resistencia al avance (Término lineal)
thetj	0.0	Ángulo del timón o del waterjet
Fm	0.0	fuerza de propulsión (N)
pmax	5000.0	Potencia máxima del motor

Figura 5.1: Interfaz de la clase barco

Ahora bien si queremos editar los atributos de la clase barco o cadena bastaría con editar sus tablas de forma interactiva y fácil para el usuario. Para crear un objeto de la clase barco o cadena, que representan la creación

Variables			Estructuras Variables Registros		
Parámetro	Valor	Explicación	Parámetro	Valor	Explicación
A	2.0	Coefficiente de resistencia al giro de un elemento	cable[1]	0.1	longitud Cable remolque derecho
q2	0.0	Coefficiente de resistencia al desplazamiento longitudinal de un elemento (Término cua...	cable[2]	1000	Constante de elasticidad
mA	0.0	Masa anadida a un elemento			
objeto	cadena	Identificador único de la barrera (nombre)			
s2	0.0	Coefficiente de resistencia al desplazamiento transversal de un elemento (Término cua...	Anch[0]	0.0	Extremo izquierdo de la barrera anclado a tierra 0/1
m	2.0	Masa de un elemento	Anch[1]	0.0	Extremo derecho de la barrera anclado a tierra 0/1
L	0.5	Semilongitud de un elemento			
q	0.0	Espesor de un elemento de la barrera	Fd[0]	0.0	Fuerza aplicada al extremo derecho de la barrera x
s	50.0	superficie de un elemento	Fd[1]	0.0	Fuerza aplicada al extremo derecho de la barrera y
est	22	Número de elementos (Eslabones)	Fil[0]	0.0	Fuerza aplicada al extremo izquierdo de la barrera x
			Fil[1]	0.0	Fuerza aplicada al extremo izquierdo de la barrera y
Guardar cadena					

Figura 5.2: Interfaz de la clase cadena

Variables			Estructuras Variables Registros		
Parámetro	Valor	Explicación	Parámetro	Valor	Explicación
tiempo	0.0	Tiempo inicial de simulación	kpid[0]	1	Constante proporcional contr...
wb	0.0	Velocidad angular (rad/s)	kpid[2]	0.1	Constante derivativa control ...
Tsf	0.1	Constante de tiempo del motor del barco	kpid[1]	1	Constante integral control PI...
M	0.0	Torque aplicado al barco por la fuerza de propulsión			
mu2	0.0	Coefficiente de resistencia al desplazamiento lateral (Término cuadrático)	kl[2]	0.0	Constante integral control PL...
mu2	0.0	Coefficiente de resistencia al giro (Término cuadrático)	kl[1]	0.0	Constante derivativa control ...
link	0	Enlace radio entre barcos	kl[0]	3	Constante proporcional contr...
theta	90	rumbo actual (rad)			
mu2	0.0	Coefficiente de resistencia al avance (Término cuadrático)	ab[0]	0.0	
Tsw	0.1	Constante de tiempo del motor del sistema de guiado	ab[1]	0.0	
cfr	100	factor de forma para los barcos guiados por timón			
mut	10000.0	Coefficiente de resistencia al desplazamiento lateral (Término lineal)	vb[1]	0.0	velocidad y
Ac	10.0	Distancia del eje del timón al centro de masa	vb[0]	0.0	velocidad x
lb	1000.0	Momento de inercia			
tipo	waterjet	Define el sistema de propulsión	Fb[1]	0.0	fuerza de externa ejercida y
alfab	0.0	Aceleración angular (rad/s ²)	Fb[0]	0.0	fuerza externa ejercida x
setl	0.0	Valor de consigna del motor			
thetwmax	30	Ángulo máximo de giro del timón o waterjet	ma[0]	0.0	Masas anadidas x
			ma[1]	0.0	Masas anadidas y
Guardar Barco					

Figura 5.3: Ampliando la explicación del atributo kpid[1]

de un USV y una barrera respectivamente, se le da *click* izquierdo en el botón verde. Botón verde que se ve en las Figuras 5.1 y 5.2 llamado 'Guardar Barco' y 'Guardar Cadena' para la interfaz de la clase barco y la interfaz de la clase cadena.

Desde ahora hablaremos de crear barco, que significará crear una instancia de clase barco y una cadena como crear una instancia de la clase cadena.

Luego de crear un barco o una cadena con un respectivo identificador(para el barco) o nombre(para la cadena). Estos nombres e identificadores son únicos y si el usuario comete el error de querer crear dos barcos o dos cadenas con identificadores o nombres iguales, la interfaz lo detectará mostrando un mensaje tal como se ve en la Figura 5.4 para la clase barco.

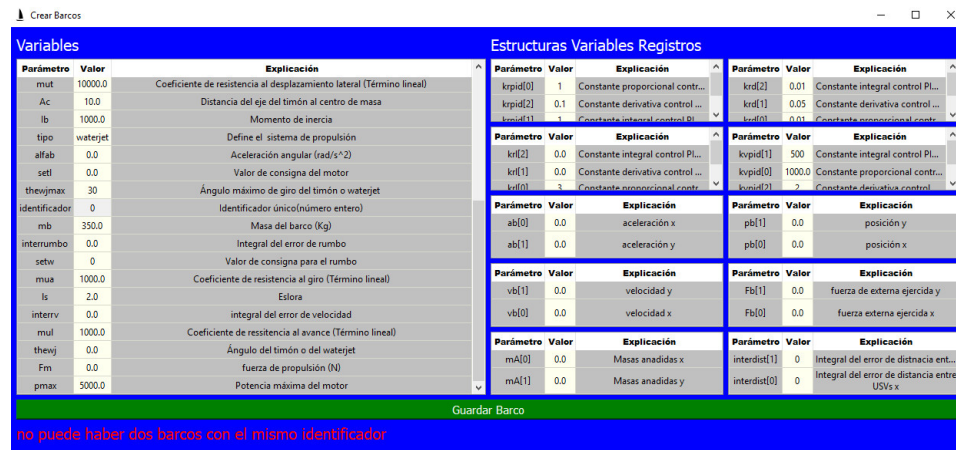


Figura 5.4: Error al querer crear dos barcos con el mismo identificador

5.2. Interfaz Principal

Después de crear la interfaz de la clase barco y la interfaz de la clase cadena, tenemos que crear una interfaz que junte a éstas dos interfaces y así conformar un escenario concreto. Haremos la interfaz para simular la acción del remolque de una barrera mediante dos USVs.

para esto debemos tener estos botones:

- Un botón para la creación de objetos de la clase barco, llamado 'Crear Barco' y otro botón para la creación de objetos de la clase cadena, llamado 'Crear Cadena'.
- Luego debemos tener un botón para la edición, que se llamará 'Editar Barcos y Cadenas', por si nos equivocamos en la creación de objetos barco o cadena y así poder corregir algún atributo mal inicializado.

- Dos botones más, uno para el enlace de las barreras y USVs mediante cables de arrastre y otro para la simulación que llamaremos: 'Combinar' y 'Simular Barco y Cadena' respectivamente.
- Un menú 'File', en el que se puede cargar simulaciones hechas anteriormente.
Al darle *clic* a este menú se despliega la opción 'Cargar Datos para una Simulación'. Opción que se usa para guardar los datos de una simulación.

Quedando el aspecto de nuestra interfaz principal como el de la Figura 5.5.

Como se puede apreciar en la Figura 5.5, los botones activos son los de 'Crear Barco' y 'Crear Cadena'. Esto es porque cuando empezamos a crear barcos y cadenas, los botones se van activando conforme vamos avanzando dentro de la interfaz.

Ahora bien cuando se crea un barco o una cadena, se activa el botón 'Editar Barcos y Cadenas'. Botón que explicaremos en la siguiente sección.

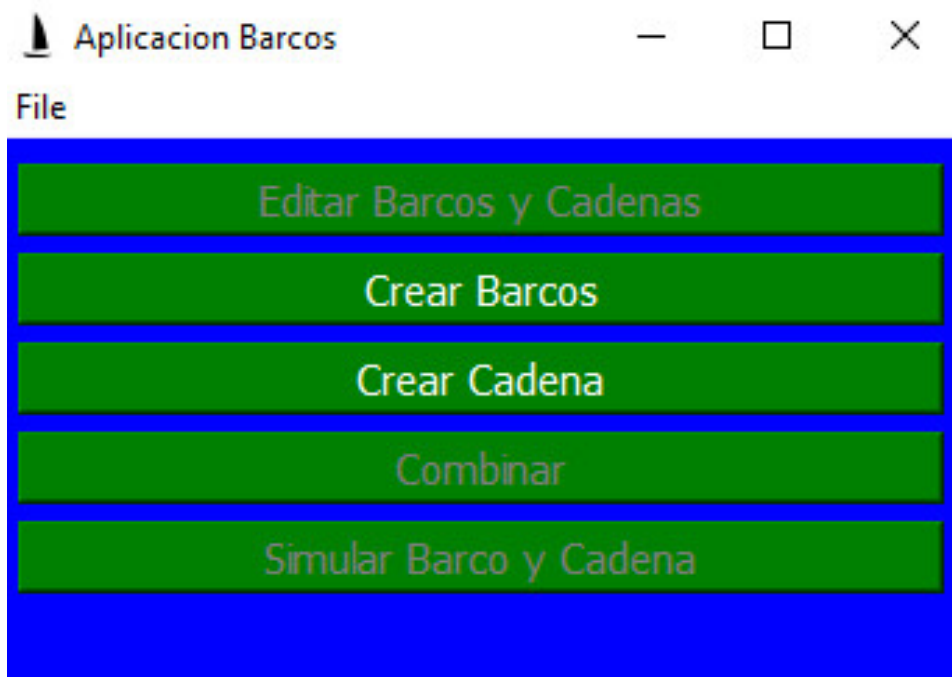


Figura 5.5: interfaz principal

5.3. Interfaz del botón Editar Barcos y Cadenas

El botón Editar Barcos y Cadenas se activa al crear uno o más barcos(USVs) o una o más cadenas(Barreras).

Ahora bien, una vez activado, como se ve en la Figura 5.6, en donde se ha creado un barco y una cadena, puedes elegir editar el barco o la cadena dándole *clic* izquierdo en el botón 'Editar Barcos y Cadenas' , que abre una ventana, tal como se ve en la Figura 5.7 después de darle *clic*.

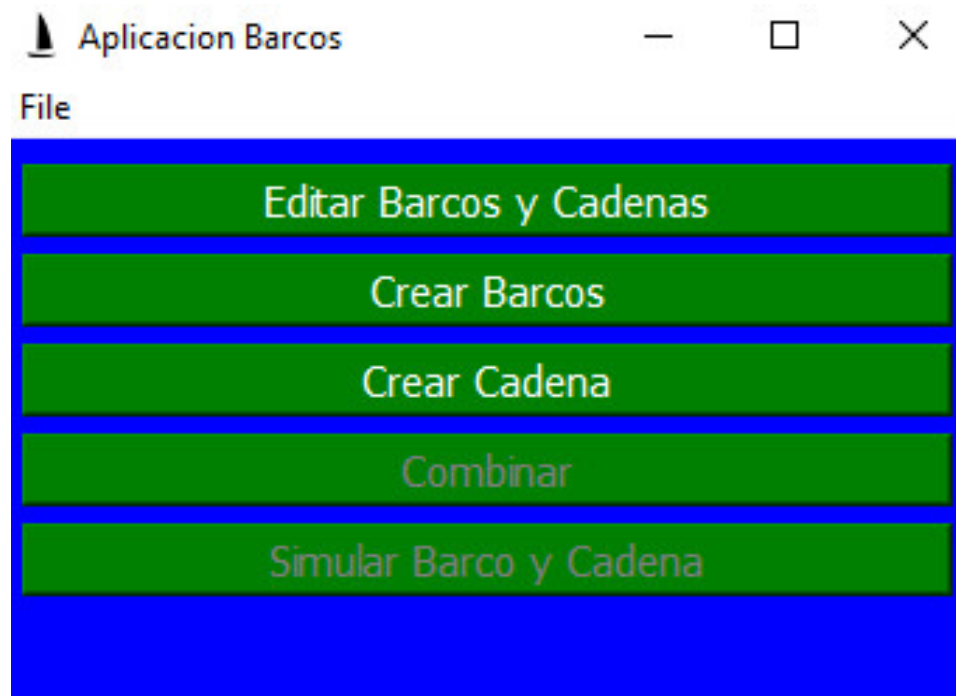


Figura 5.6: interfaz de Edición de Barco o Cadena

Una vez escogido lo que se quiere editar, que es bien un barco o una cadena, la interfaz que se muestra es parecida a la interfaz barco o cadena antes mostrada, pero con la diferencia de que ahora hay un botón más llamado 'Mostrar Dibujo Barco' el cual muestra una imagen del barco o la cadena tal como se aprecia en la Figura 5.8.

En la Figura 5.9 se puede ver la figura de un barco que es lo que te muestra el botón 'Mostrar Dibujo Barco' después de haberlo *cliqueado*.

Luego de crear al menos dos barcos y al menos una cadena se activa el botón 'Combinar' que explicaremos en la siguiente sección.

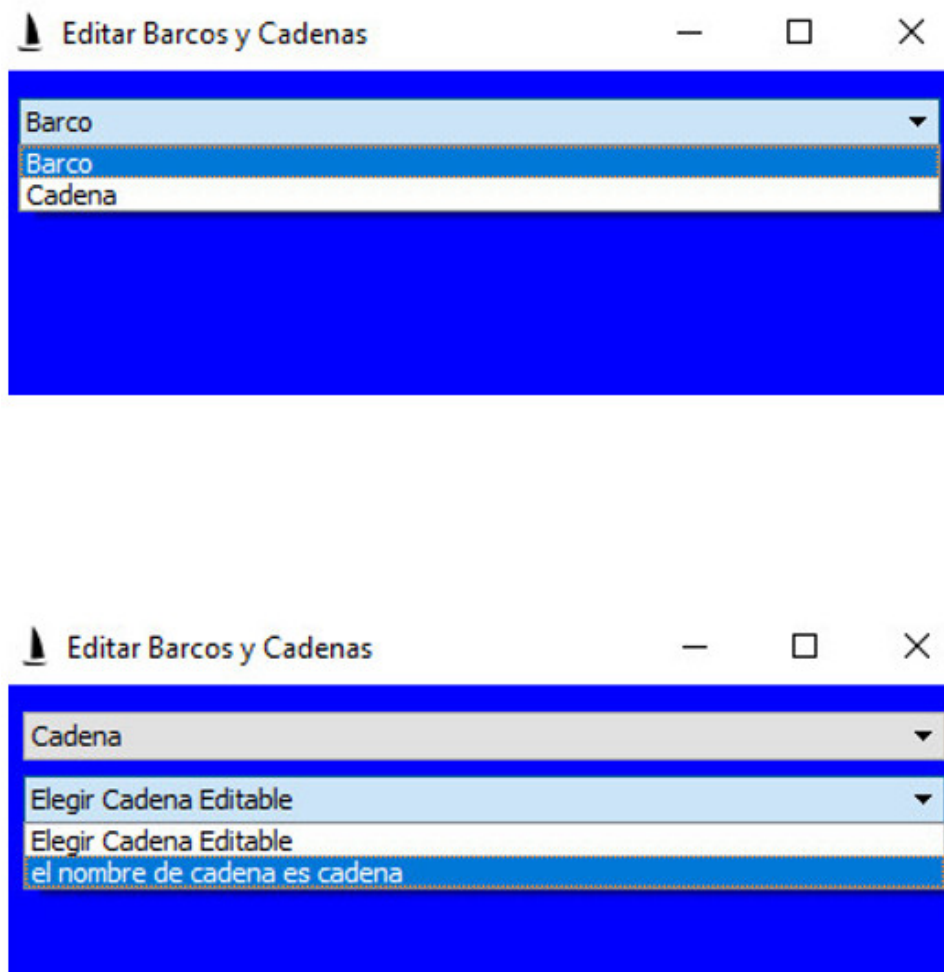


Figura 5.7: Dos pantallas de la interfaz Editar Barcos y Cadenas

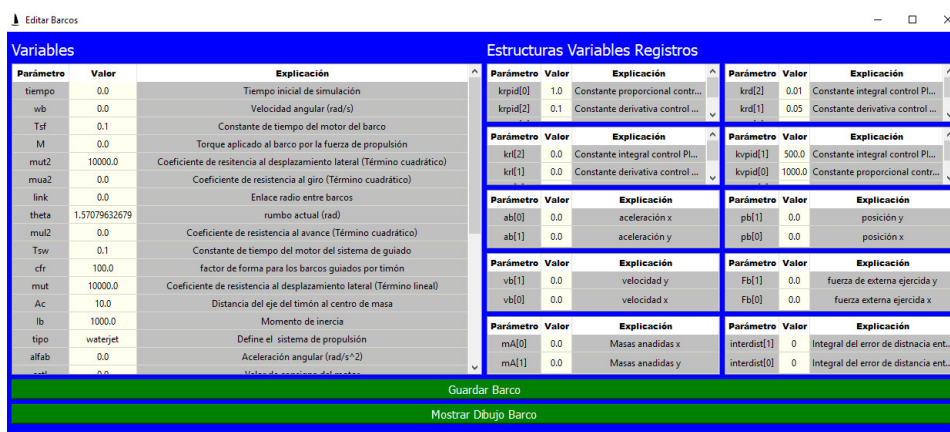


Figura 5.8: Muestra el botón añadido : Mostrar Dibujo Barco

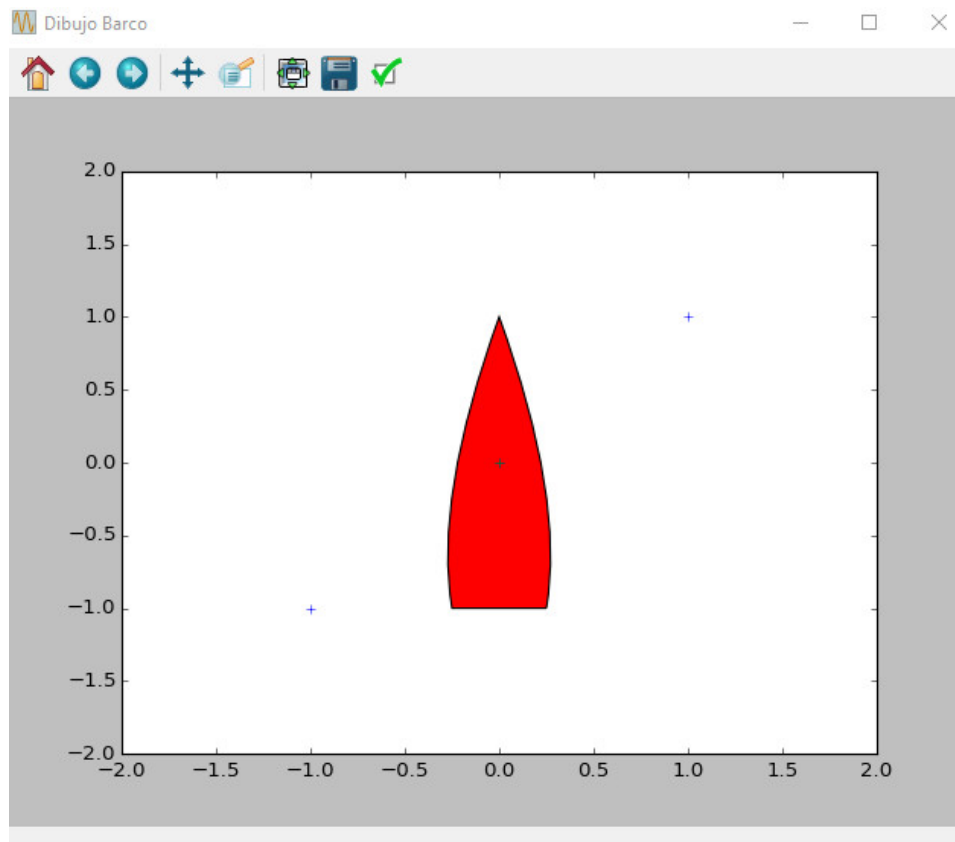


Figura 5.9: El dibujo del barco(USV)

5.4. Interfaz de Combinar

Una vez activado y *cliqueado* el botón, se abre una ventana para poder enlazar los barcos con la cadena. Ésta ventana tiene un total de: Tres botones y tres desplegables como muestra la Figura 5.10. Estos botones y desplegables se describen a continuación.

- El botón desplegable elige el barco que usaremos para la simulación, en el segundo desplegable escogeremos la cadena y por último en el tercer desplegable se elige el otro barco que usaremos en la simulación. En la Figura 5.11, vemos que hemos elegido el barco con identificador '0', una cadena llamada 'cadena' y un barco con identificador '1'.
- El botón elongación te deja elegir que porcentaje de la longitud del cable, que une el barco derecho(o izquierdo) con el último eslabón del extremo derecho(o izquierdo) de la cadena, se aplicará al inicio de la simulación, así si le ponemos un porcentaje alto el cable irá más suelto inicialmente y si por el contrario le ponemos un porcentaje bajo el cable irá inicialmente más tenso. Para esto cuando le damos *clic* se abre una ventana la cual te deja elegir la cantidad en % que puedes escoger. Un ejemplo de esto lo muestra la Figura 5.12 en donde vemos que se ha escogido el 80 %.
- El botón Radio Enlace , que se nota en la Figura 5.11, te abre una ventana en la cual escogemos los atributos que vamos a comunicar de barco(con identificador '0') a barco(con identificador '1') para hacer el control cooperativo. Para esto cuando le damos *clic* se abre una ventana en donde se pueden seleccionar a los atributos que elijamos para el control cooperativo de los barcos. Un ejemplo de esto lo muestra la Figura 5.13, en donde elegimos los atributos ide y theta. Una vez elegido los atributos a comunicar, le damos *clic* al botón 'Guardar Radiolink' como se ve en la Figura 5.13.
- El botón Enlazar se activa después de la elección anterior. Al darle *clic* en éste botón se enlaza los barcos con la cadena que van a remolcar y se cierra la ventana.

Por último se activa el botón 'Simular Barco y Cadena' que explicaremos a continuación.

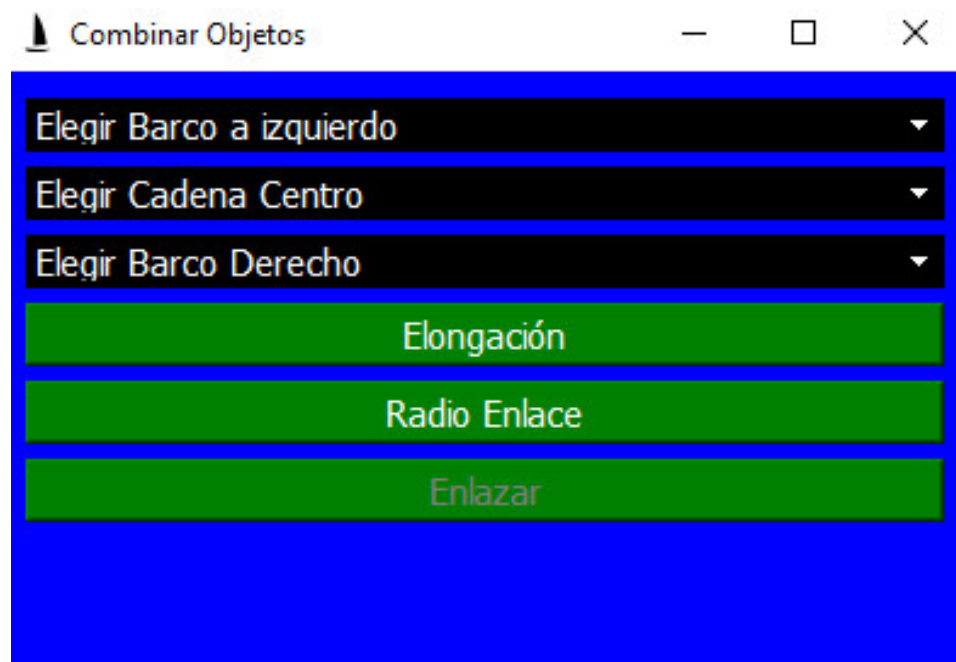


Figura 5.10: Interfaz del botón Combinar

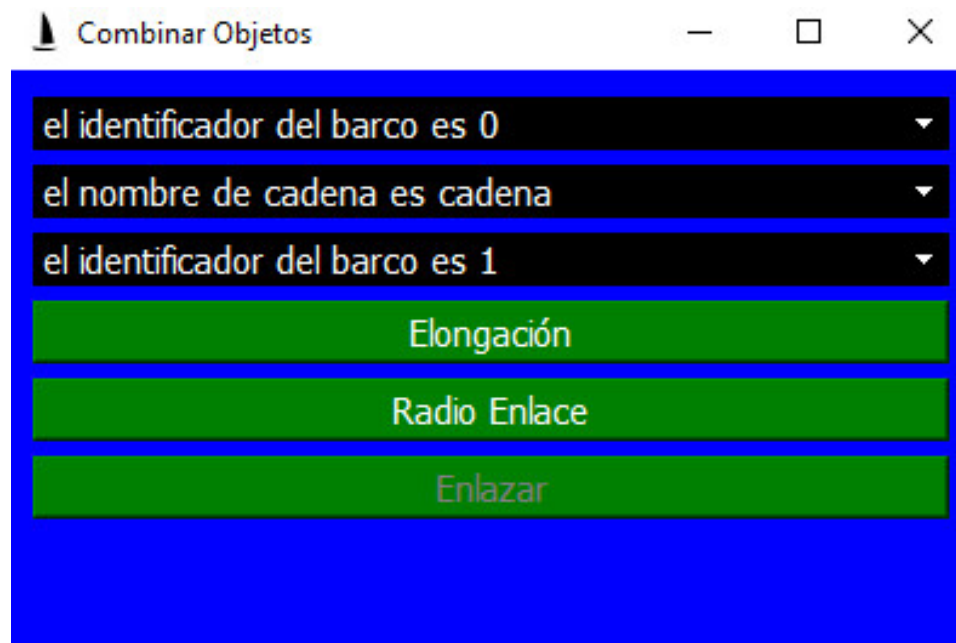


Figura 5.11: Interfaz del botón Combinar cuando se escoge el barco y la cadena

5.5. Interfaz del botón Simular Barco y Cadena

Una vez activado y *cliqueado* el botón, se abre una ventana que muestra una tabla de seis botones, un desplegable y una barra de carga. A continuación se explican todos estos elementos:

- La tabla muestra los atributos que caracterizan el entorno simulado en donde se va a realizar la acción del remolque de la barrera mediante los dos USVs. Estos atributos son los que debemos configurar cada vez que se hace una simulación.

Como se aprecia en la Figura 5.14 la tabla Tiene tres columnas, que muestran los atributos de forma análoga al de las tablas de barco o cadena explicadas anteriormente.

El atributo 'step' de Ésta tabla se usa para calcular el total de iteraciones que se van a realizar, el cual se calcula como la división $\frac{tam}{step}$ siendo 'tam' el tamaño del experimento, y el atributo 'delta' el paso de integración elegido. Los demás atributos se abordarán con más detalle en el siguiente capítulo.

- El botón 'Cargar' inicia el proceso de carga de los datos de la simulación para cada iteración, el cual tiene un tiempo de simulación largo, ya que el programa calcula los valores de todas las tensiones en los extremos de los eslabones para cada iteración.

Este tiempo depende del tamaño de la barrera que se desea remolcar, el tiempo de simulación total y del paso de integración que se emplee. En la Figura 5.17, mostramos el momento en que el botón 'cargar' ha sido *cliqueado* y se está procesando los datos para la simulación.

- El botón 'run' inicia el proceso en el que se muestra el esquema total de simulación, una vez terminado el proceso de carga de los datos, explicado en el párrafo anterior.

El esquema total de simulación es la unión del esquema de cada iteración. El esquema de cada iteración es la representación de los valores de los datos obtenidos en cada iteración. Dicho esquema representa el estado del USV y la barrera para los datos de una iteración en concreto. En la Figura 5.15 se muestra el momento en que el botón 'run' se ha *cliqueado* y el proceso ha avanzado 8 %.

- El botón 'stop', se usa para parar procesos, tanto el proceso de carga de los datos como del proceso en el que se muestra el esquema total de simulación.

En la Figura 5.16 se vé cómo se para el proceso de carga de los datos y qué es lo que muestra la ventana al momento de pulsar el botón 'stop'.

- El desplegable 'Elegir el Modo Pasos', despliega tres opciones:
 1. normal
Este modo recoge todas las iteraciones para mostrarlas en un desplegable llamado 'Pintar Estado'(que se explicará en el siguiente elemento),
 2. intercalado 10 veces
Este modo recoge las iteraciones intercaladas cada 10, para mostrarlas en un desplegable 'Pintar Estado'(que se explicará en el siguiente elemento).
 3. intercalado 100 veces
Lo mismo que en el párrafo anterior salvo porque se intercala cada 100.

En la Figura 5.18 se vé tanto el desplegado y también el momento en que escogemos una opción en el desplegable 'Elegir el Modo Pasos'.

- El botón 'run estado' abre una ventana en donde se muestra un desplegable llamado 'Pintar Estado' que contiene las iteraciones escogidas en el elemento anterior.
En este desplegable se puede seleccionar un estado de la simulación(que es un esquema de una iteración en concreto)que al darle *clic* a la opción del desplegable llamado 'Button (número de iteración) Estado' se muestra el esquema de la iteración elegido(estado de la simulación).
Un ejemplo de esto lo vemos en la Figura 5.19.
- El botón 'Gráficos Adicionales' nos muestra los gráficos:
 1. Tensión en los cables de arrastre.
 2. Velocidad
 3. Distancia
 4. Rumbo

En la Figura 5.20 se vé tanto el desplegado, para escoger cual de los gráficos mostrar y uno de los gráficos mostrados.

- El botón 'Save' guarda los datos de simulación obtenidos una vez acabada la simulación. Algunas simulaciones completas se explicarán con más detalle en la siguiente sección.

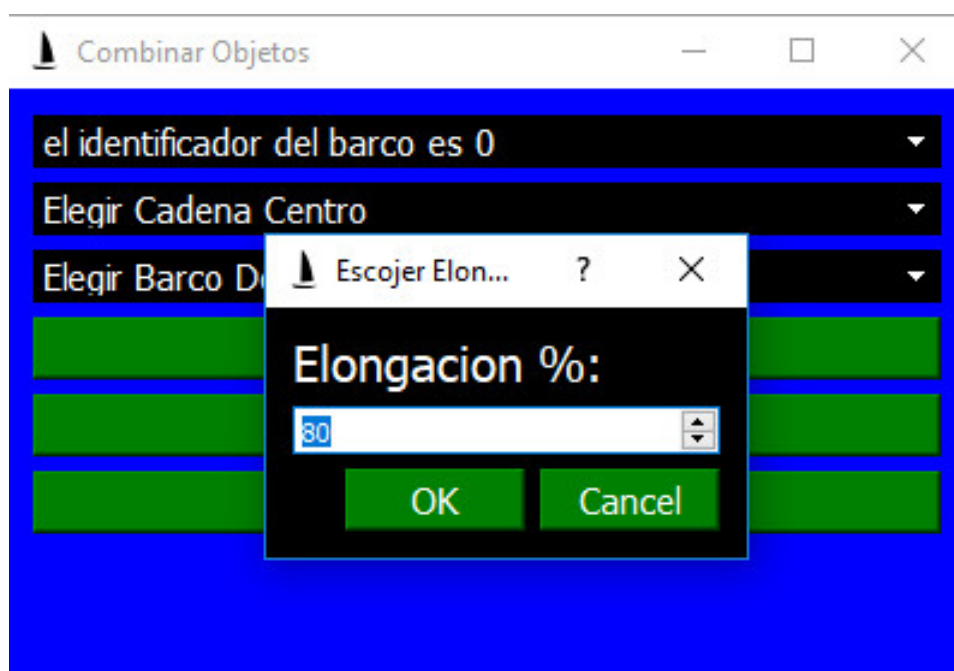


Figura 5.12: interfaz del botón Elongación

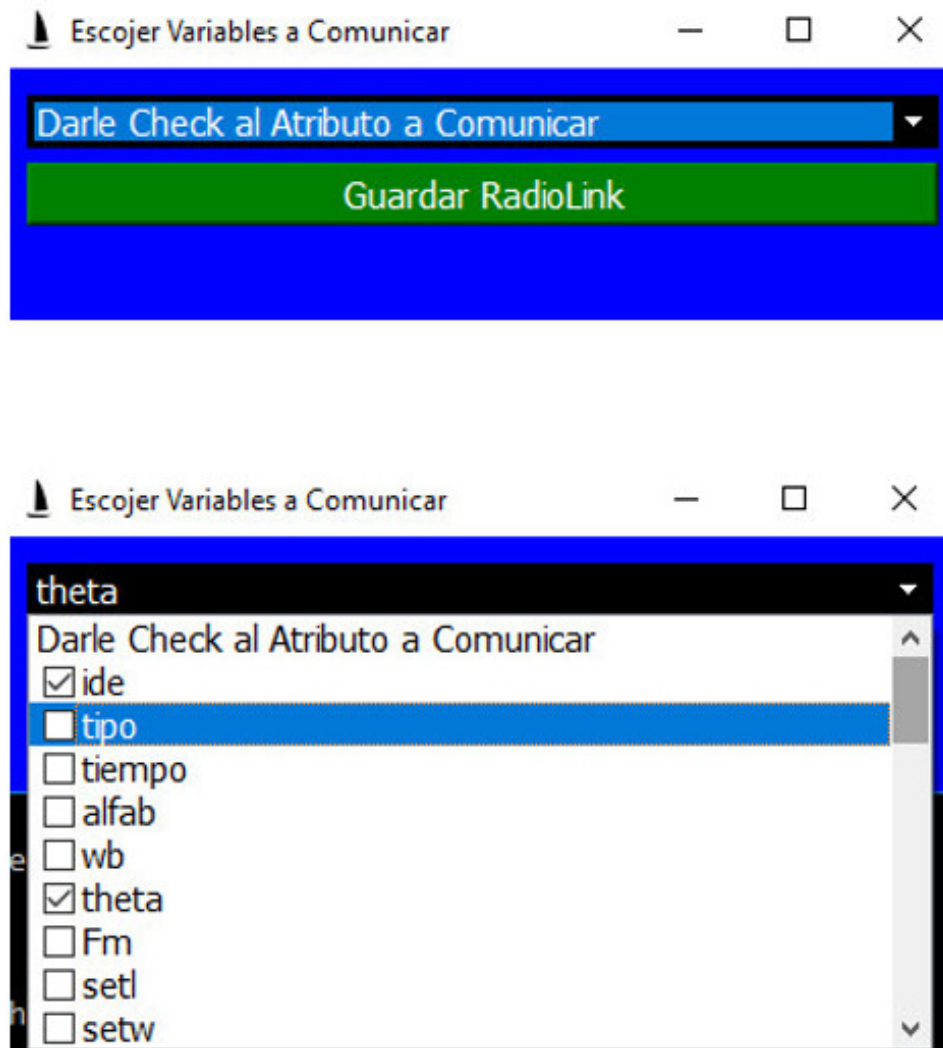


Figura 5.13: dos ventanas del botón Radio Link, en donde se puede observar que hemos escogido los atributos ide y theta

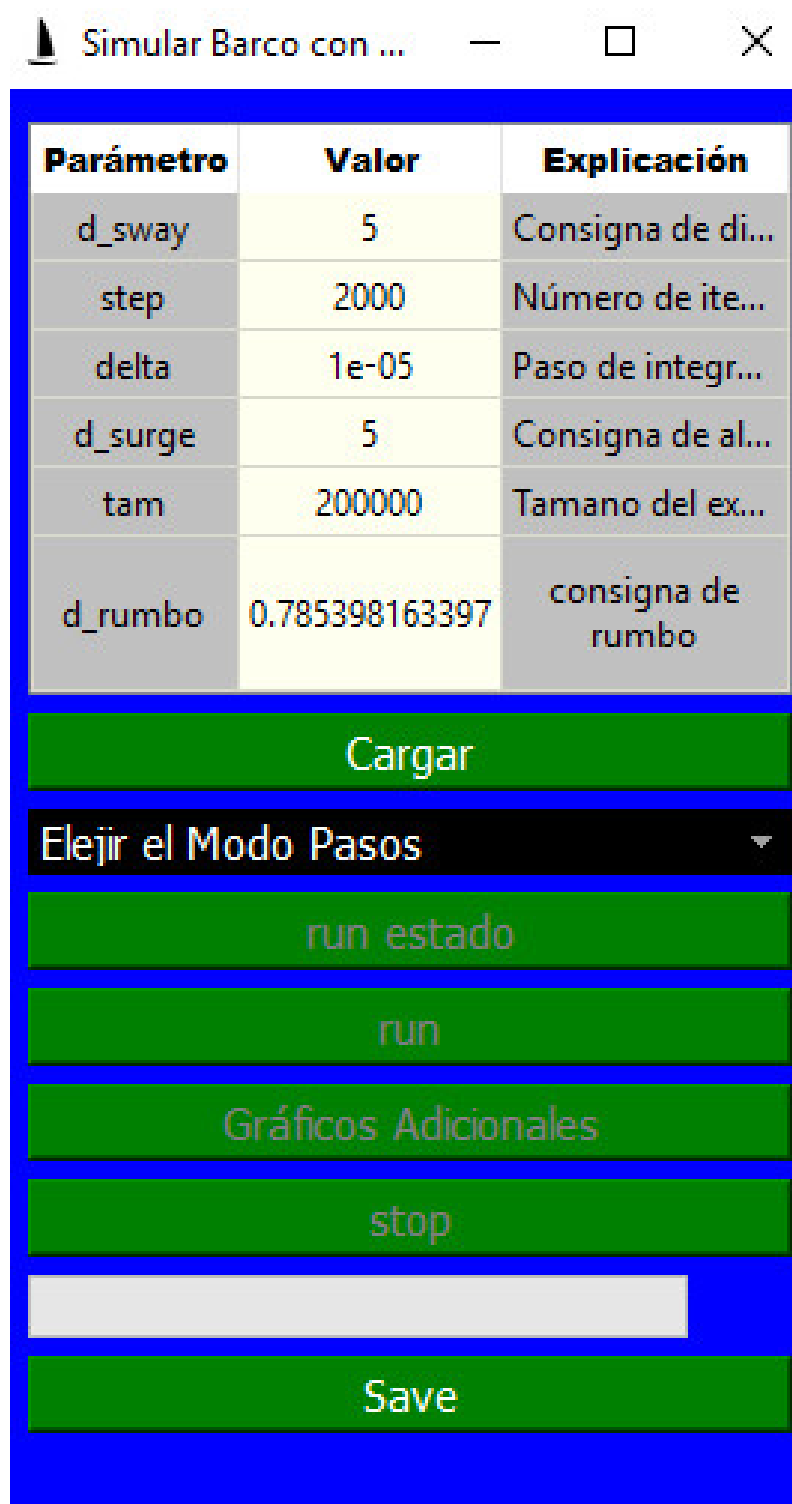


Figura 5.14: Interfaz del botón Simular Barco y Cadena

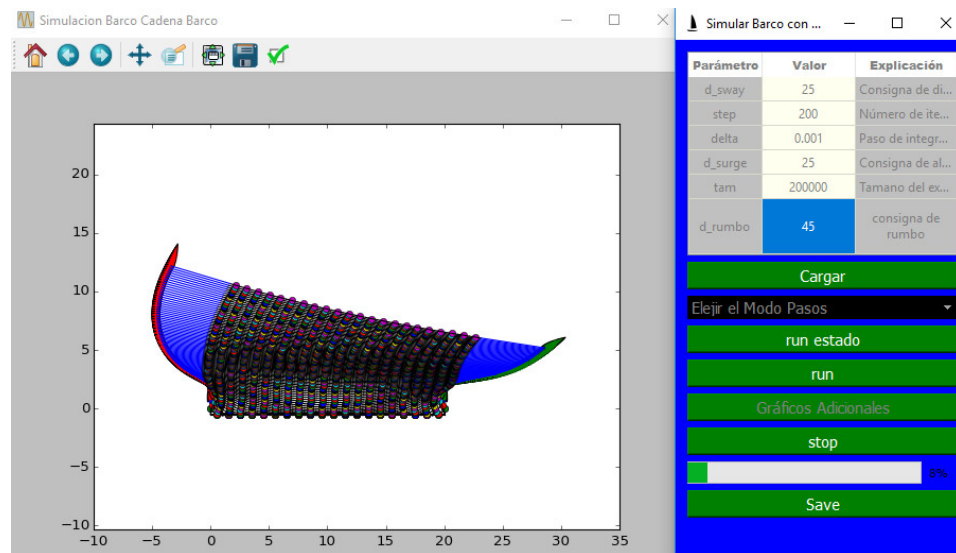


Figura 5.15: Simulación con 0.001 de paso de integración y una consigna rumbo de 45 grados

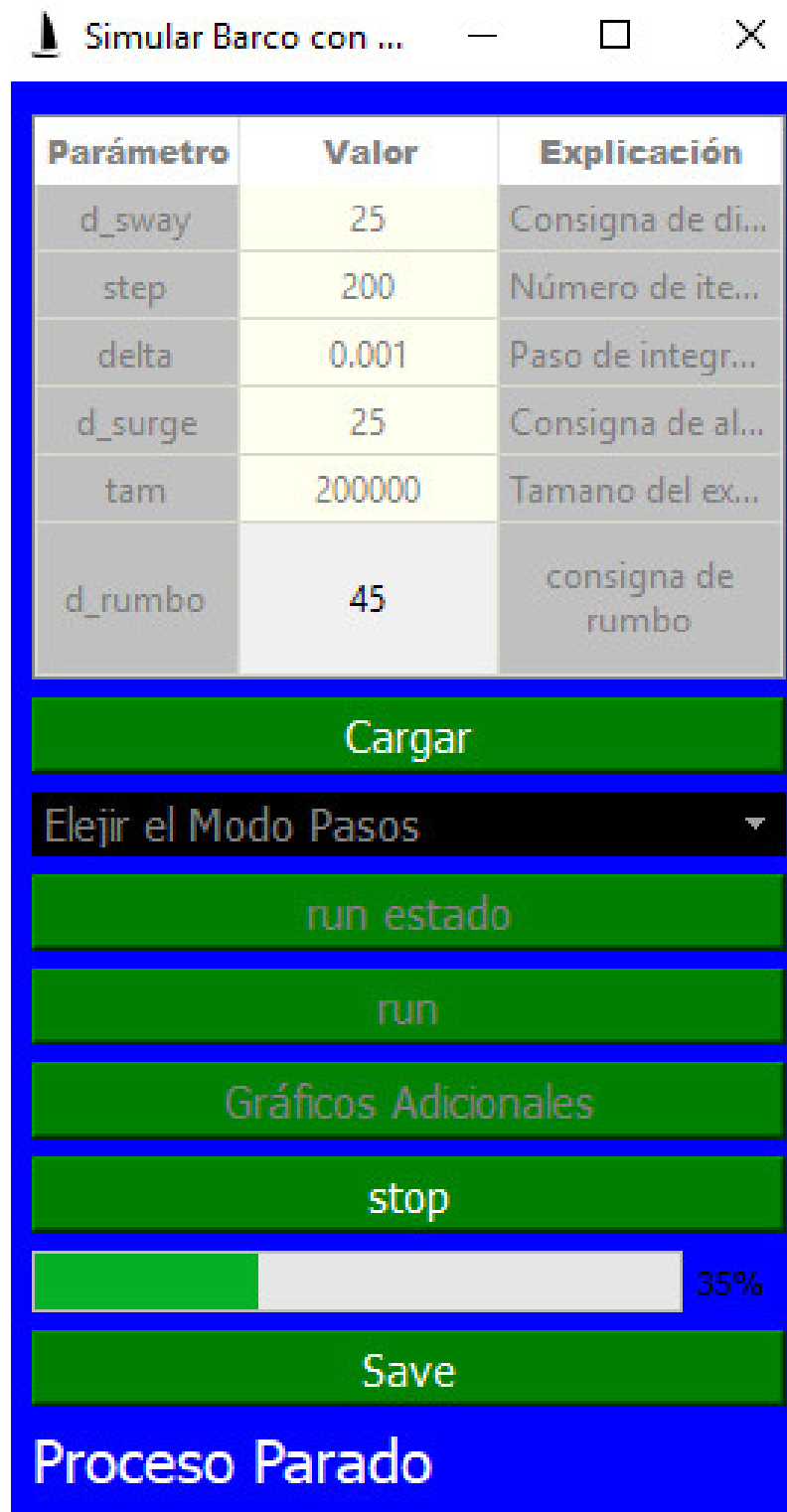


Figura 5.16: Parado el proceso de carga de datos

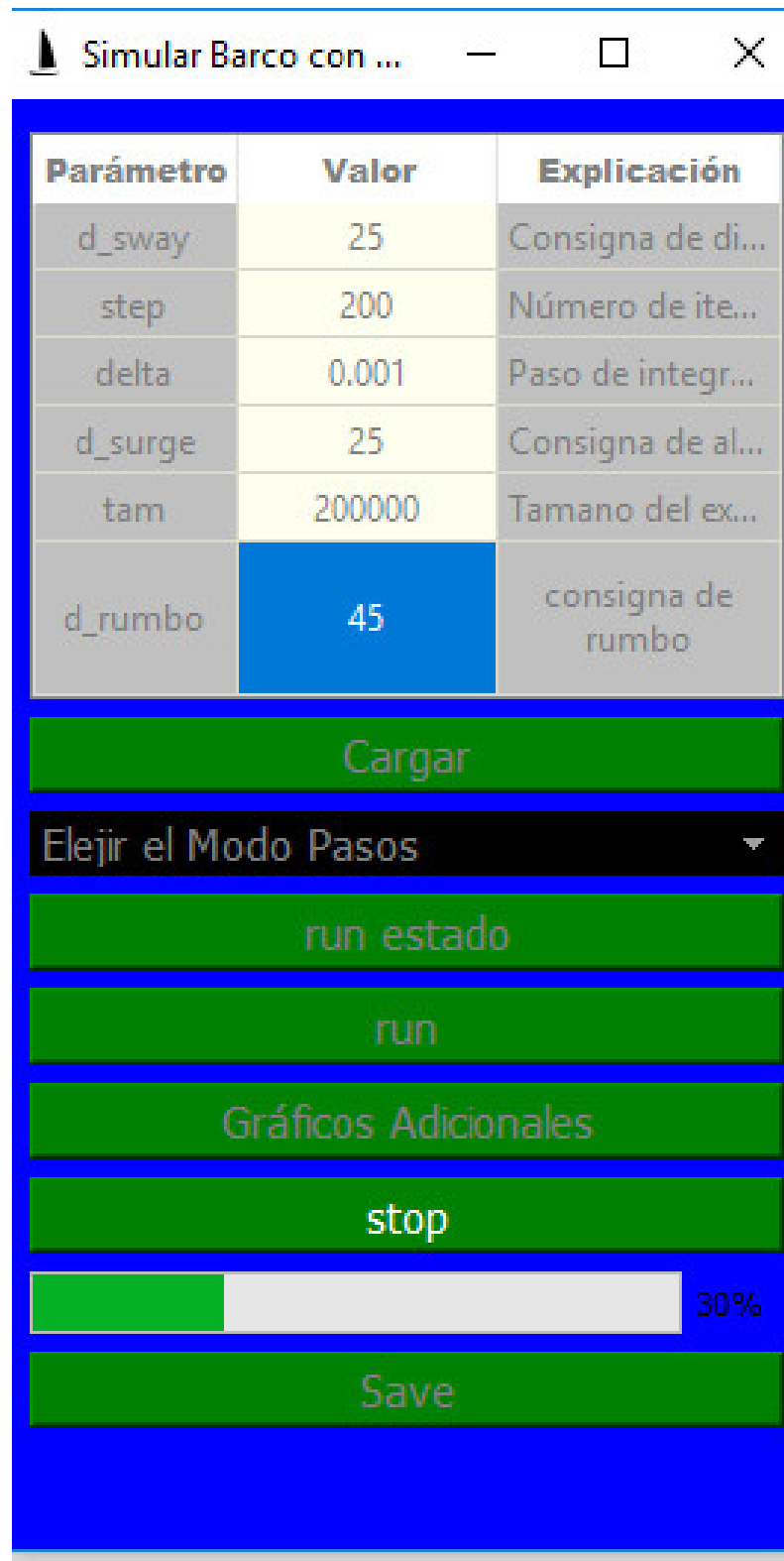


Figura 5.17: Cargando datos para la simulación

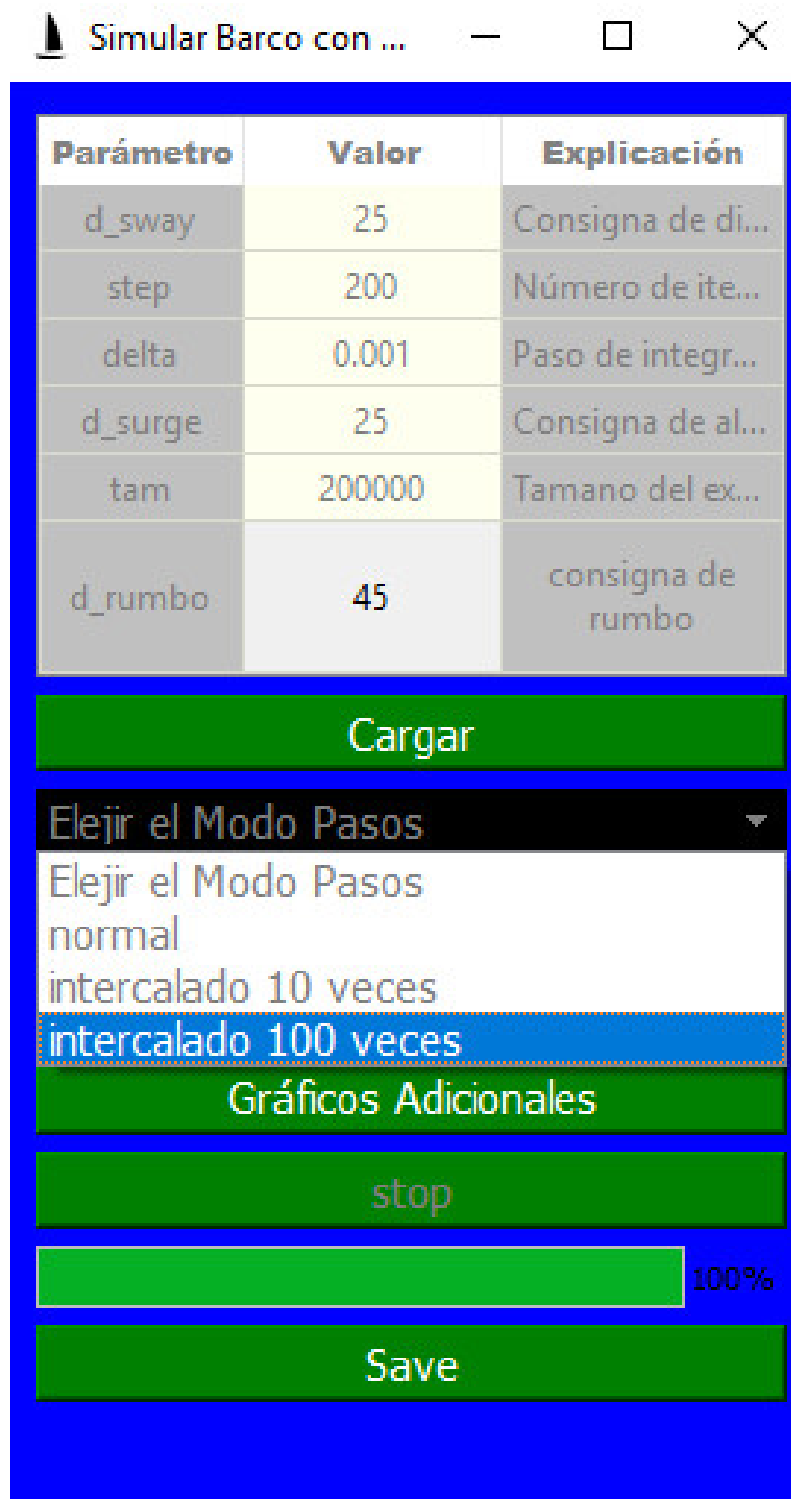


Figura 5.18: Eligiendo un grupo de iteraciones

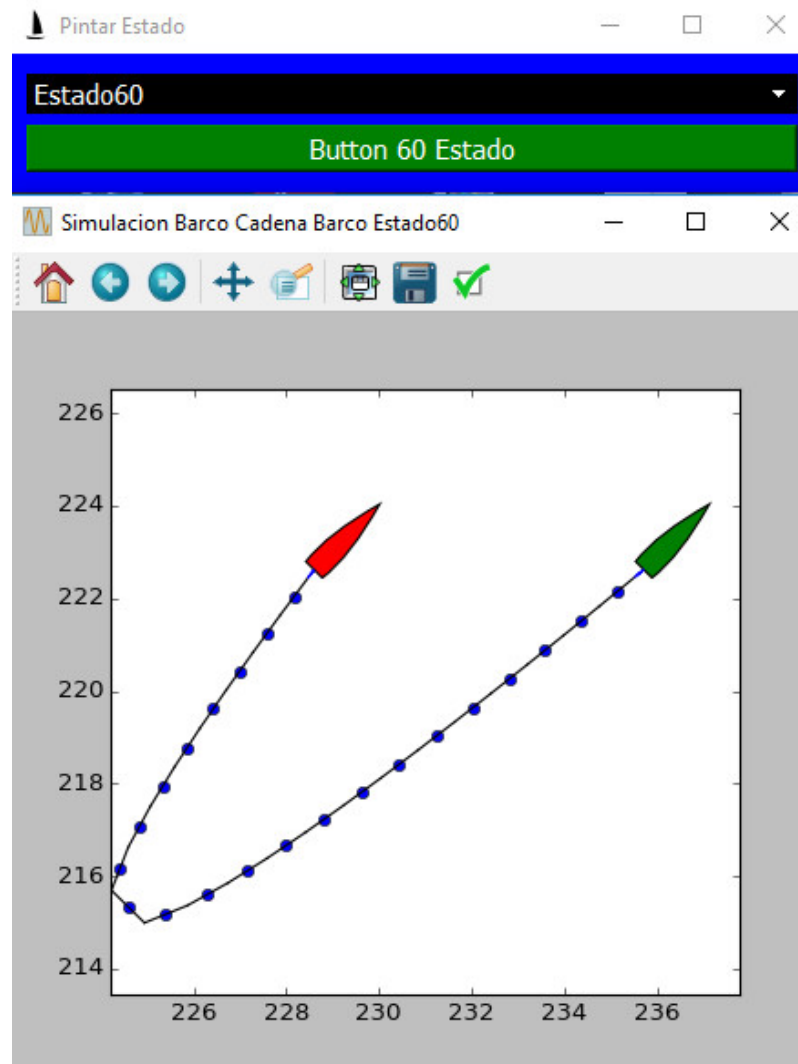


Figura 5.19: Elegimos la iteración 60 y luego de hacer *clic* se muestra el esquema de ésta iteración

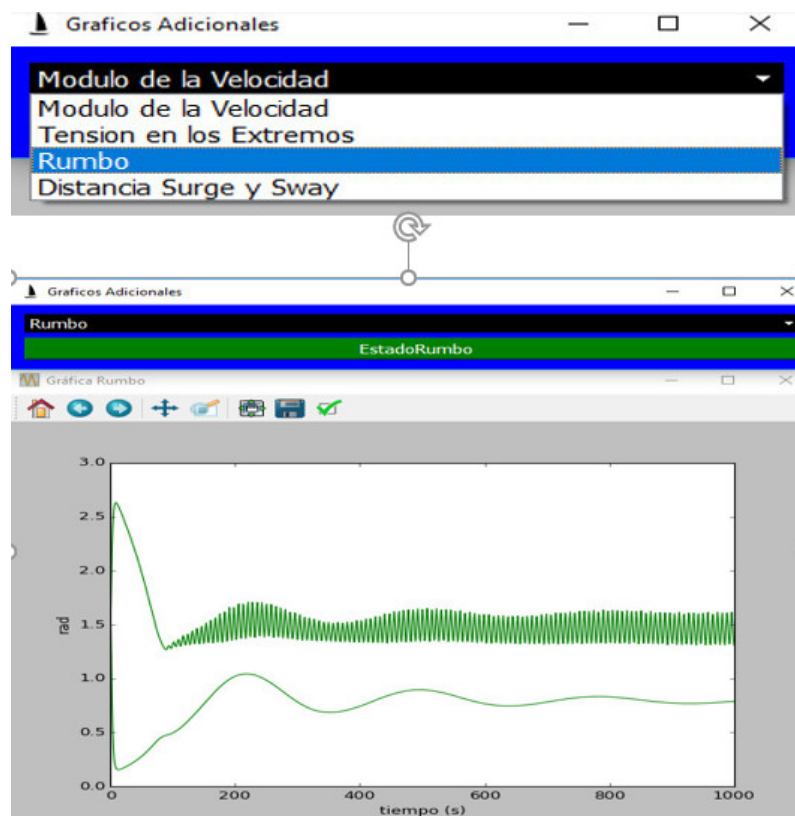


Figura 5.20: Se muestra la elección de la gráfica Rumbo y la gráfica Rumbo

Capítulo 6

Simulación Completa

Vamos a realizar una simulación completa y mostrar como se representa el funcionamiento del sistema cooperativo que se explicó en el capítulo 3.4. Para esto primero iniciamos la interfaz principal(Figura 5.5). *cliqueamos* los botones 'Crear barcos' y 'Crear Cadena' de manera que tengamos dos instancias de la clase barco y una instancia de la clase cadena, configurando ésta última con 200 eslabones(en la tabla de la Figura 5.2 es el atributo 'esl') y un cable de 5 metros en ambos extremos de los barcos y una elongación de 1000(en la tabla de la Figura 5.2 son los atributo cable[0],cable[1],cable[2] respectivamente), los demás atributos se dejan por defecto. En cuanto a los barcos se les dá identificadores distintos para ambos barcos(en la tabla de la Figura 5.1 es el atributo 'identificador'),los demás atributos se dejan por defecto. Luego como explicamos en el capítulo 5.4 enlazamos los barcos con la cadena, comunicando cuatro variables entre los barcos con 'El botón Radio Enlace', éstas variables son:

1. La variable 'ide' que me indica el identificador del barco.
2. la variable 'theta' que me indica el rumbo del barco.
3. la variable 'pv' que me indica la posición del barco.
4. la variable 'vb' que me indica la velocidad del barco.

Luego se encenderá el botón 'Simular Barco y Cadena' como se explicó en el capítulo 5.4 y así configuraremos los parámetros de la tabla de ésta interfaz('Simular Barco y Cadena').

Estos parámetros son:

- El parámetro 'd_sway' te dá la consigna de distancia lateral en metros que llegan a tener ambos barcos para seguir su rumbo. En nuestro caso le asignamos 20 metros.

- El parámetro 'step' que es el número de iteraciones con el que se calcula el total de iteraciones que se van a representar durante la simulación. En nuestro caso le asignamos 2000.
- El parámetro 'delta' te da el paso de integración numérica que se empleará en el experimento, cuanto más pequeño sea éste, el movimiento se describirá con más detalle y el tiempo de simulación aumentará como explicamos en el capítulo 5.5. Para nuestro caso le asignaremos '0.001'.
- El parámetro 'd_surge' te da la consigna de alineamiento en metros que llegan a tener ambos barcos para seguir su rumbo. Para nuestro caso le asignamos 20 metros.
- El parámetro 'tam' te da el tamaño numérico total del experimento. Para nuestro caso le asignamos 200000 para que tengamos una cantidad suficiente de iteraciones con las que describir mejor el movimiento.
- El parámetro 'd_rumbo' te da el valor del rumbo medido en radianes (pero que se edita en grados en la tabla) que van a seguir ambos barcos una vez alineados con ambas consignas. Para nuestro caso le asignaremos 45 grados.

Ahora bien una vez configurada la tabla anterior, le damos *clic* al botón 'cargar', y luego como se explicó en el capítulo 5.5 se terminará el proceso y le daremos *clic* al desplegado 'Elegir el Modo Pasos' y escogemos el intercalado de 10 y 100 tal como se explicó en el capítulo 5.5. Luego vemos la Figura 6.1 que muestran las gráficas de la simulación al completo. Como se ve en la Figura 6.1(a) los USVs se van acercando hasta llegar a la distancia de alineación establecida (d_{surge}) y girán hasta adecuarse a su distancia lateral (d_{sway}) para seguir su rumbo. Ahora bien en la Figura 6.1(a) se ve más o menos lo que hemos dicho, pero en la Figura 6.1(c) se nota mejor y en la Figura 6.1(b) se nota el inicio y como terminarían siguiendo su rumbo. Luego podemos terminar de analizar ésta simulación dándole al botón 'Gráficas Adicionales' que muestra 4 gráficas las cuales vemos en la Figura 6.2. En las gráficas que se ven en la Figura 6.2 se muestra la evolución de las variables más relevantes durante el proceso de arrastre de la barrera con los dos USVs. Explicaremos cada una de éstas gráficas:

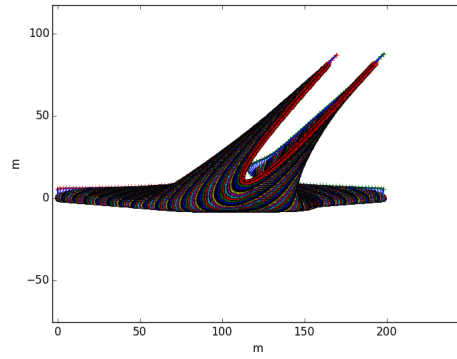
- La gráfica 6.2(d), representa la tensión que soporta los cables, durante toda la simulación, para los dos USVs. Ésta gráfica se compone de dos líneas una verde, que representa la tensión del cable que está enganchado al USV derecho, y otra roja, que representa la tensión del cable que está enganchado al USV izquierdo. Ahora bien en ésta gráfica se

nota que durante los primeros 4 segundos los USVs tienen el mismo valor de tensión, luego oscilan de manera compensada de entre 4 a 6 segundos, es decir uno va ganando cierto valor de tensión y el otro va perdiendo mas o menos la misma tensión. Esto sería porque en esos segundos los USVs se van alineando. A partir de los 6 segundos en adelante van avanzando con distinta tensión, esto sería porque ambos arrastran la barrera soportando distintas tensiones por el remolque de distintos trozos de barrera.

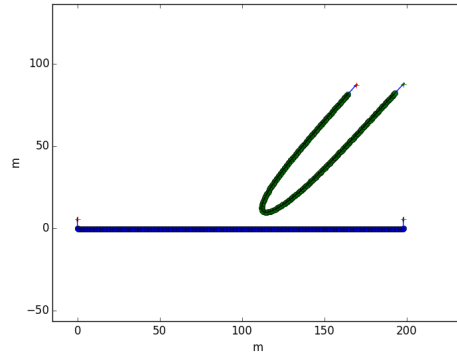
- La gráfica 6.2(a) representa la velocidad en ambos USVs, en donde la linea roja es para el USV izquierdo y la verde para el derecho. Como se nota en ésta gráfica y se notó en la gráfica anterior, en los primeros 4 segundos los USVs van avanzando mas o menos con la misma velocidad pero entre los 4 y 6 segundos oscilan otra vez de manera compensada, ya que están maniobrando para alinearse. Y a partir de los 6 segundos van a velocidad constante de 1m/s.
- La gráfica 6.2(b) representa el rumbo o dirección de los USVs, en donde la linea roja es para el USV izquierdo y la verde para el derecho. Se nota que en los primeros 4 segundos el USV derecho rota hasta los π radianes y el otro hasta los 0 radianes(en la gráfica 6.1(c) se nota se nota esto). Luego entre los 4 y 6 segundos van adquiriendo el rumbo deseado hasta que a partir de los 6 segundos van con el rumbo indicado.
- La gráfica 6.2(c) muestra los valores de alineación representada por las variables `d_surge` y `d_sway` de ambos USVs respecto al tiempo. Aquí vemos que los valores van decreciendo linealmente en los primeros 4 segundos para luego oscilar, esto sucede durante las maniobras que hacen para la alineacion y luego a partir de los 6 segundos llegan a la alineación deseada(20 metros).

Ahora bien, después de ésta simulación podríamos crear otra a partir de donde nos quedamos(Figura 6.1(b)). Osea una vez que los USVs están arrastrando a la barrera con el rumbo indicado. Solo tendríamos que cambiar los valores de los atributos de la interfaz 'Simular Barco y Cadena' sin tener que cerrar la ventana. Entonces configuramos los parámetros con estos valores:

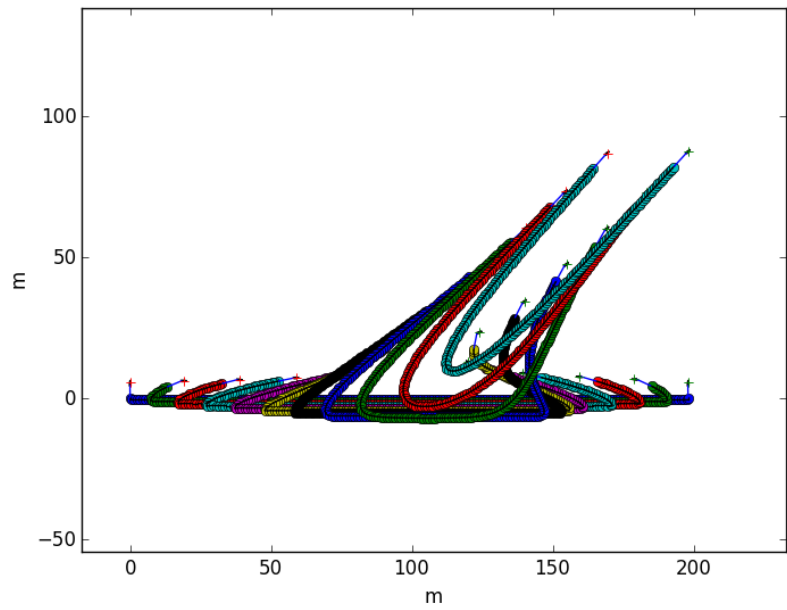
- Al parámetro '`d_sway`' le asignamos 50 metros.
- Al parámetro '`step`' le asignamos 2000.
- Al parámetro '`delta`' le asignamos '`0.001`'.
- Al parámetro '`d_surge`' le asignamos 50 metros.



(a)

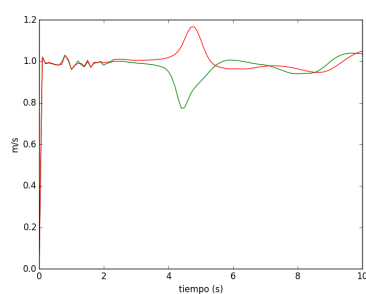


(b)

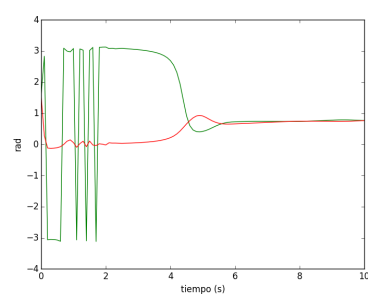


(c)

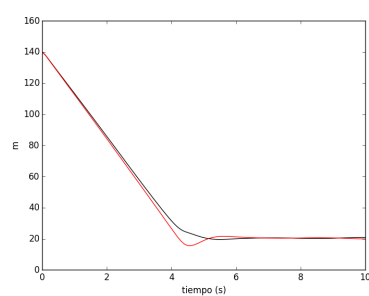
Figura 6.1: Dos USVs arrastrando una barrera de 200 metros unidos mediante un cable de 5 metros, una consigna de rumbo de 45 grados y una velocidad final de $1m/s$ y un tiempo de 10 segundos de simulación.



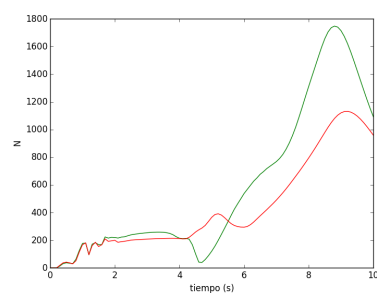
(a)



(b)



(c)



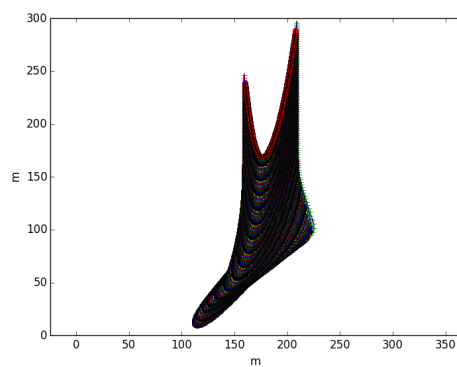
(d)

Figura 6.2: Gráficas de la simulación completa

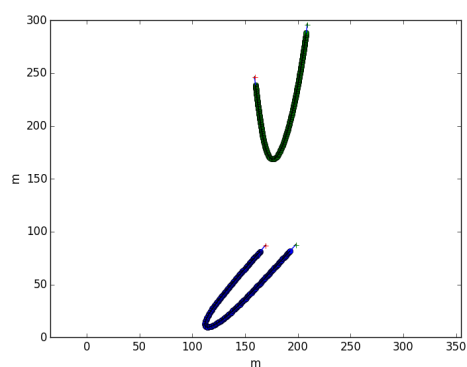
- Al parámetro 'tam' le asignamos 200000.
- Al parámetro 'd_rumbo' le asignamos 90 grados.

Se hace lo mismo que en la simulación anterior obteniendo las gráficas de la Figura 6.3 y la Figura 6.4. Vemos que ahora se tienen que alinear 50 metros y tienen que tener un rumbo de 90 grados. Ésta evolución se ve en la gráfica 6.3(a) , 6.3(c) y se nota el inicio y el final en la gráfica 6.3(b). Luego en las gráficas:

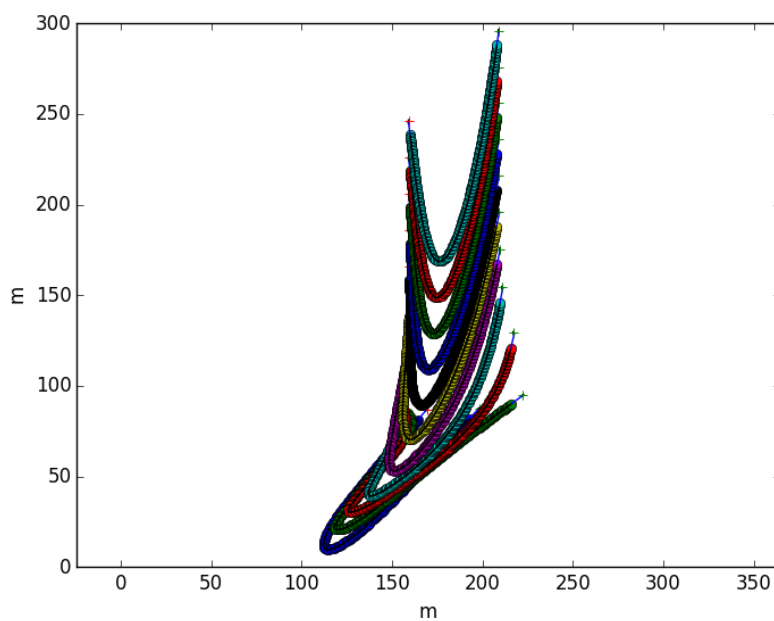
- En la gráfica de tensión 6.4(d) se ve que en los primeros 4 a 5 segundos se van repartiendo las tensiones para llegar a la nueva alineación y rumbo.
- En la gráfica 6.4(a) de la velocidad, en los primeros 4 a 5 segundos la velocidad oscila a partir de 1m/s pero se normaliza después de los 5 segundos a 1m/s.
- En la gráfica 6.4(b) del rumbo, vemos que el USV izquierdo(la línea roja) gira desde $\pi/4$ hasta cerca de π para luego girar de nuevo hasta llegar a $\pi/2$ y el barco derecho en concordancia con este gira desde $\pi/4$ hasta cerca de cero radianes para luego girar hasta $\pi/2$. Y esto pasa en los primeros 4 a 5 segundos para luego seguir con el rumbo indicado.
- En la gráfica 6.4(c) de alineación, se nota igual que en las anteriores gráficas, que en los primeros 4 a 5 segundos oscilan para luego normalizarse en los siguientes segundos.



(a)



(b)



(c)

Figura 6.3: Dos USVs arrastrando una barrera de 200 metros unidos mediante un cable de 5 metros, a partir de una consigna de rumbo de 45 grados hasta llegar a una consigna de rumbo de 90 grados y una velocidad final de $1m/s$, en un tiempo de 10 segundos de simulación.

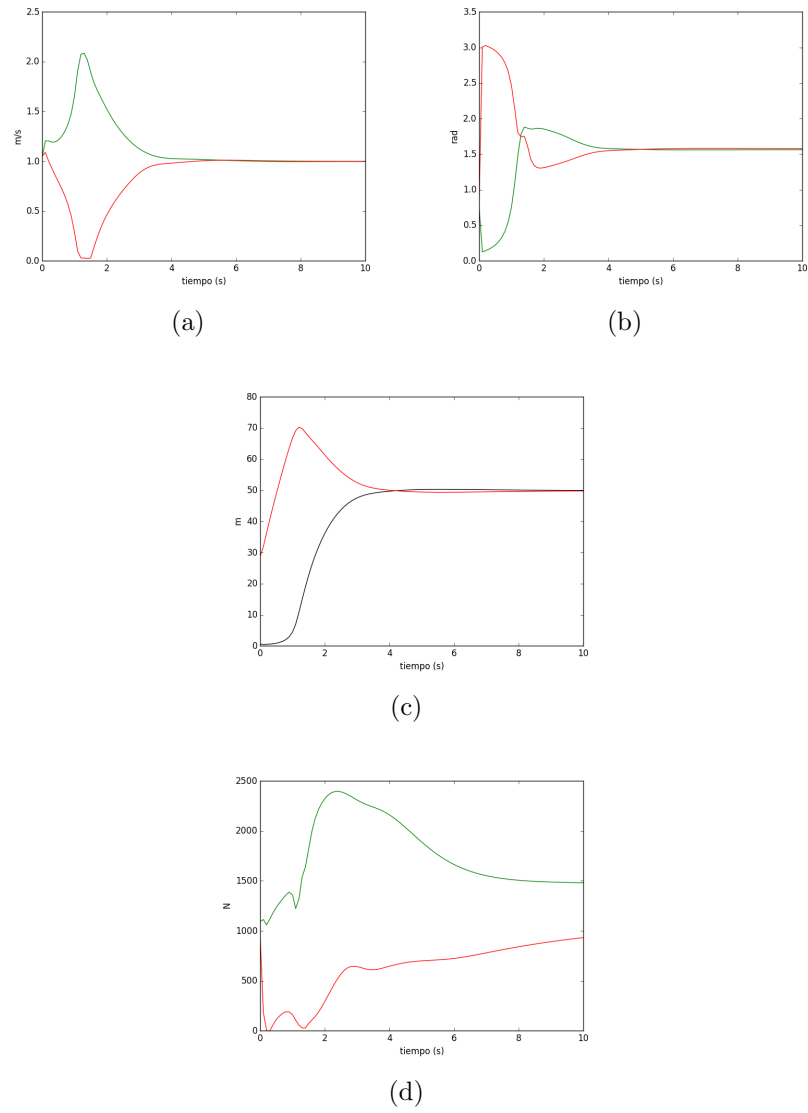


Figura 6.4: Gráficas de la siguiente simulación completa siguiendo la simulación completa anterior

Capítulo 7

Implementación de la Interfaz.

Para la implementación de la Interfaz usamos Pyqt4 como explicamos en el capítulo anterior. La librería Matplotlib la usamos para los Gráficos del movimiento de remolcado de la barrera con los USVs. Así como También para los gráficos de Rumbo, Velocidad, Distancia y Tensión.

La clase MainWindow tiene las siguientes funciones, que resuelven una serie de tareas, tal como explicamos a continuación:

- función Cadenacrear:
Ésta función crea una instancia de la clase cadena 'Clase MyCadena', que representa una barrera. Ésta función es la que se encarga de manejar los eventos en la 'Interfaz Cadena' de la clase cadena.
- función BarcoCrear:
Ésta función crea una instancia de la clase barco 'Clase MyBarco', que representa un USV. Ésta función es la que se encarga de los eventos de la 'Interfaz Barco' de la clase barco.
- función editarbarcoycadena:
Ésta función crea una 'Clase MyEditaByC' que manipula los objetos barco o cadena ya creados en las funciones anteriores . Ésta interfaz es la que se encarga de los eventos de la Interfaz 'Edita Barco y Cadena'.
- función Combinar:
Ésta función crea una 'Clase MyCombinar' que se encarga de unir a dos instancias barcos y una instancia cadena, Preparando así a los objetos para la simulación. Ésta interfaz es la que se encarga de los eventos de la 'Interfaz Combinar'.

- función simular:
Ésta función crea una 'Clase MySimula' cuya principal tarea es crear cinco objetos contenedores llamados:
 1. lista_barcos
 2. lista_cadenas
 3. lista_barcoD_Simu
 4. lista_barcoI_Simu
 5. lista_cadenaC_Simu
 6. lista_simulacion

Para poder ejecutar todas las iteraciones de la integración, y por cada iteración, guardar los datos para la simulación.

Para poder crear las diferentes interfaces aquí mostradas, se usó Qwidgets de PyQt4:

- QComboBox
Con este Widget hacemos los desplegables que se usan en las distintas ventanas que conforman la interfaz creada.
- QHBoxLayout y QVBoxLayout
Con estos Widgets se alinean todos los demás widget y así presentar una ventana ordenada y grata al usuario.
- QPushButton
Con este Widget se crean los distintos botones que se usan en las distintas ventanas que conforman la interfaz creada.
- QLabel
Con este Widget se crean los avisos de error mostrados en las distintas ventanas que conforman la interfaz creada.
- QMutex y pyqtSignal
Estos Widgets se usan para realizar la concurrencia. Se usa concurrencia para parar los procesos, de lo contrario, se bloquearía la ventana principal y no se podría hacer nada en la interfaz cuando se está ejecutando algún proceso.
- QFileDialog
Este Widget se usa para manipular los ficheros usados para cargar o guardar.

- **QRect**
Este Widget se utiliza para configurar el tamaño y forma rectangular de las distintas ventanas de la interfaz.
- **QIcon**
Este Widget se usa para añadir un ícono a la ventana de la interfaz.
- **QTableWidget**
Este Widget se usa para crear y manipular las distintas tablas que se usan en las distintas ventanas que conforman la interfaz creada.

Capítulo 8

Conclusión y futuros trabajos

- Con ésta interfaz se mejora el uso de la herramienta para crear distintas simulaciones ya que no se necesita pensar en como está implementada sino que tenemos casi toda la configuración a un *clic* para hacerlas.
- El usuario de la interfaz tiene el manejo de todos los atributos tanto de los USVs como de las barreras, ya que se pueden ver en una tabla que se edita con solo *cliquear* en la celda correspondiente.
- Con ésta interfaz se puede hacer simulaciones seguidas sin perder el hilo, es decir de que la interfaz es capaz de recordar el último estado de los USVs y la barrera de tal manera que a partir de ahí se puede hacer una nueva simulación.
- Las gráficas de una simulación están ordenadas de manera que se puede ver cada estado en el que se encuentra los USVs y la barrera a lo largo del tiempo de simulación.
- Las gráficas se pueden ordenar con intercalado de manera que se puede llegar a ver una simulación con espacios para ver mejor el movimiento de los USVs con la barrera durante el remolcado de ésta.
- Ésta Interfaz podría aumentar el uso de la herramienta para que más adelante se tengan datos tal que se pueda crear un diseño real.
- Se podrían añadir más escenarios a la interfaz, por ejemplo el remolcado de una barrera con un solo barco o con más de dos barcos.
- Se podría mejorar el proceso de recogida de datos usando mejor la concurrencia.

CONCLUSIONS AND FUTURE WORK

- With this interface the use of the tool is improved to create different simulations because it is not necessary to think about how it is implemented because we have almost all the configuration to a *clic* to make them.
- The user of the interface has the management of all the attributes of the USVs and the Booms, because you can see it in a table that is edited by *cliquing* on the corresponding cell.
- With this interface you can do simulations in a row without losing the thread, ie the interface is able to remember the last state of the USVs and the Boom in such a way that a new simulation can be made from there.
- The graphs of a simulation are arranged so that you can see each state in which the USVs are located and the Boom over the simulation time.
- The graphics can be requested with collation so you can see a simulation with spaces to better see the movement of the USVs with the Boom during towing.
- This interface could increase the use of the tool so that the data is later available, so that a real design can be created.
- More scenarios could be added to the interface, for example, the towing of a Boom with a single USV or with more than two USVs.
- The data collection process could be improved using better concurrency.

Bibliografía

- [1] ARRICHIELLO, F., S. CHIAVERINI, AND T. FOSSEN. 2006. “FORMATION CONTROL OF MARINE SURFACE VESSELS USING THE NULL-SPACE-BASED BEHAVIORAL CONTROL”. IN GROUP COORDINATION AND COOPERATIVE CONTROL, EDITED BY K. Y. PETTERSEN ET AL., 1–19. BERLIN, HEIDELBERG: SPRINGER.
- [2] GIRON-SIERRA, J. M., A. T. GHEORGHITA, G. ANGULO, AND J. F. JIMENEZ. 2015. ‘PREPARING THE AUTOMATIC SPILL RECOVERY BY TWO UNMANNED BOATS TOWING A BOOM: DEVELOPMENT WITH SCALE EXPERIMENTS’. OCEAN ENGINEERING 95:23–33.
- [3] JIMENEZ, J. 2016. NOTES ON THE DEVELOPMENT OF A MODEL FOR SPILL BOOM TOWING. TECHNICAL REPORT, UNIVERSIDAD COMPLUTENSE DE MADRID.
- [4] [HTTPS://WWW.ELESPANOL.COM/ESPANA/20181113/GUARDIACIVIL-DETECTAVERTIDOSPRIMERAOPERACIONMUNDIAL352965083_0.HTML](https://www.elespanol.com/ESPANA/20181113/GUARDIACIVIL-DETECTAVERTIDOSPRIMERAOPERACIONMUNDIAL352965083_0.html)
- [5] [HTTPS:ES.WIKIPEDIA.ORGWIKIDERRAME_DE_PETR%C3%B3leo](https://es.wikipedia.org/wiki/Derrame_de_petr%C3%B3leo)
- [6] MODELLING THE AUTOMATIC DEPLOYMENT OF OIL-SPILL BOOMS: A SIMULATION SCENARIO FOR SEA CLEANING DECEMBER 2018 DOI: 10.1109/WSC.2018.8632255 CONFERENCE: 2018 WINTER SIMULATION CONFERENCE (WSC)
- [7] [HTTPS:ES.WIKIPEDIA.ORGWIKIPYQT](https://es.wikipedia.org/wiki/PyQt)
- [8] [HTTPS:ES.WIKIPEDIA.ORGWIKIQT_\(BIBLIOTECA\)](https://es.wikipedia.org/wiki/Qt_(biblioteca))
- [9] [HTTPS:ES.WIKIPEDIA.ORGWIKISCIpy](https://es.wikipedia.org/wiki/SciPy)
- [10] [HTTPS:ES.WIKIPEDIA.ORGWIKINUMPY](https://es.wikipedia.org/wiki/NumPy)
- [11] [HTTPS:ES.WIKIPEDIA.ORGWIKIMATPLOTLIB](https://es.wikipedia.org/wiki/Matplotlib)